

VŠB – Technická univerzita Ostrava
Fakulta strojní
Katedra automatizační techniky a řízení

Podpora experimentální výuky Základů automatizace
Experimental Support for Teaching of Control Systems
Fundamentals

Student: David Kadlec

Vedoucí bakalářské práce: doc. Ing. Renata Wagnerová, Ph.D.

Ostrava 22.05.2009

Prohlášení studenta

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením vedoucího bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě:

.....

podpis studenta

Prohlašuji, že

- byl jsem seznámen s tím, že na moji diplomovou (bakalářskou) práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména §35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a §60 – školní dílo.
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě diplomovou (bakalářskou) práci užít (§35 odst. 3).
- souhlasím s tím, že jeden výtisk diplomové (bakalářské) práce bude uložen v Ústřední knihovně VŠB-TUO k prezenčnímu nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové (bakalářské) práce. Souhlasím s tím, že údaje o diplomové (bakalářské) práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu §12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou (bakalářskou) práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě:.....

.....

podpis studenta

Anotace

Kadlec David

Podpora experimentální výuky základů automatizace

Ostrava: Katedra automatizační techniky a řízení, Fakulta strojní

VŠB–Technická univerzita Ostrava, 2009, 32s. Bakalářský projekt,

Vedoucí: Wagnerová R.

Bakalářská práce se zabývá metodikou podpory experimentální výuky Základů automatizace a to konkrétně prostřednictvím zařízení od firmy ABB. Jedná specificky o dvě demo a to demo typu 800xA a demo dávkovacího systému. V úvodu jsou popsány jejich hardwarové komponenty a softwarové vybavení.

Práce se dále věnuje především popisu a způsobu programování jejich PLC. Její praktická část se dále zabývá odstraněním komunikačního nedostatku na demo 800xA a následným ověřením jeho funkčnosti. Poté jsou na základě získaných znalostí vytvořeny podrobné návody pro práci s oběma těmito modely dem.

V závěru jsou následně popsány a zhodnoceny dosažené výsledky.

Die Anotation

Kadlec David

Unterstützung für die experimentelle Lehre der Grundlagen der

Automatisierung, Ostrava: Institut für Automatisierungs-Technologie

und Steuerung, Fakultät für Maschinenbau, VŠB-Technische Universität

Ostrava, 2009, 32s. Bachelores Projekt, Leiter: R. Wagnerová

Bachelore Arbeit wird sich mit der Methodik der experimentellen Unterricht Fundamento Automatisierung befasst und speziell mit den Geräte von Firma ABB. Es gibt speziell um zwei Demo und konkret ist das Demo des Typs 800xA und Demo des dosierenden Systems. In der Einführung werden seine Hardware-Komponenten und Software beschreibt.

Die Arbeit wird sich vor allen mit der Beschreibung und mit dem Art der Programmierung seines PLC befasst. Ihre praktische Teil wird sich weiter mit der Entfernung des kommuniken Mangel bei Demo des Typs 800xA und mit der nachgeschalteten Bestätigung seiner Funktionalität befasst. Dann werden auf der Grundlage der Wissen die detaillierten Anweisungen für den Umgang mit diesen Demos entwickelt.

Am Ende werden am folgend die erzielten Ergebnisse beschreibt und bewertet.

Obsah

Seznam použitého značení, zkratk a odborných termínů	7
1 Úvod	10
2 Popis vybraných laboratorních modelů	11
2.1 Demo 800xA	11
2.1.1 Procesorový modul AC 800M (typ PM860)	11
2.1.2 Modul CI854A	12
2.1.3 Vstupní/výstupní analogový modul AO801	13
2.1.4 Modul diskretních vstupů V/V DI801	13
2.1.5 Výstupní modul DO801	14
2.1.6 Napájecí zdroj SD821	15
2.1.7 Frekvenční měnič ACS800-01/U1	15
2.1.8 Ovládací panel PLC PP220	17
2.2 Demo dávkovacího systému	21
2.2.1 Elektromagnetický průtokoměr FXE4000, PROFIBUS DP	22
2.2.2 Tlakový senzor 265 GS, Hart	22
2.2.3 Teplotní senzor TF 12 / TF 12-Ex	24
2.3 Softwarové vybavení uvedených dem	24
3 Popis PLC AC 800M a způsob jeho programování	26
3.1 Programovatelný logický automat AC 800M	26
3.2 Základní programovací struktura	27
4 Ověření funkčnosti dema 800xA	29
5 Návod pro práci s modelem dávkovacího systému	36
6 Návod pro práci s modelem ABB 800xA	42
7 Závěr	47
Použitá literatura	49

Seznam použitého značení, zkratek a odborných termínů

Značení

A	Ampér – jednotka elektrického proudu
A/D	analogově digitální převodník
AC 800M	procesorový modul
ACS800-01/U1	frekvenční měnič
AO801	vstupní/výstupní analogový modul PLC
B	Byte - jednotka vyjadřující velikost paměti
b/s	(bit za sekundu) jednotka přenosové rychlosti
CI854A	komunikační modul PLC
CN1, CN4	komunikační porty PLC
DC	označení stejnosměrného proudu či napětí
DI801	modul diskrétních vstupů PLC
DO801	modul diskrétních výstupů PLC
DSV 401	programové rozhraní teplotního senzoru, ve kterém lze přenastavovat jeho parametry
DTC	metoda přímé regulace točivého momentu střídavých motorů
FI2214	průtokový snímač
FXE 4000, PROFIBUS DP	elektromagnetický průtokoměr pracující na průmyslové sběrnici Profibus DP
H2210	ohřívač
HV2260	vypouštěcí ventil
Hz	Hertz – jednotka frekvence
J405	druh ovládacího panelu frekvenčního měniče
LI2201	snímač hladiny
LIC2201	kontroler hladiny nádrže
LV2201	ventil řízený PID regulátorem
MPC860	mikroprocesor PLC AC 800M
PI2201 a PI2202	dvoutlakové snímače
PM 860	typ procesorového modulu AC 800M
PM2205	přítoková pumpa
PT100	teploměr

RS-485	standard, ze kterého vychází průmyslová sběrnice
RS – 232C	porty PLC používané pro dvoubodovou komunikaci
SD821	napájecí zdroj, který dodává energii samotným jednotkám PLC a to i vstupním a výstupním modulům
TF 12/TF 12EX	teplotní senzor
TI2210	teplotní senzor
V	Volt – jednotka napětí
W	Watt – jednotka výkonu či příkonu
XV2240	přítokový ventil
XV2250	odtokový ventil
265 GS, Hart	tlakový senzor
192 I/O	signál, který se připojuje přes elektrickou sběrnici PLC
1344 I/O	signál, který se připojuje přes optickou sběrnici PLC

Zkratky

CPU	central processing unit – procesorová jednotka PLC
FLASH	paměť, na kterou lze zapisovat informace a opětovně je zase vymazat bez vlivu zapnutého či vypnutého zdroje
LED	svítivá dioda
PID	proporcionálně integračně derivační analogový regulátor
PLC	programovatelný logický automat
PROM	paměť, ze které už nelze opětovně vymazat nahrané informace
SDRAM	paměť pracující při stejném taktu, jaký je nastaven na paměťové sběrnici
LCD	(liquid crystal display) – plochá obrazovka složená z tekutých krystalů
LPT	paralelní port
COM1,2,3	sériový port

Odborné termíny

Aspect Project	aspekt, který slouží k popisu objektu
Demo	soubor zařízení složený z programovatelného logického

	automatu, jeho komunikačních prvků a přídatných modulů
Demo Dual Screen Operator Workplace	operátorské pracoviště rozděleno do dvou obrazovek (displejů) na počítači
Draining	sekvence vypouštění
ControlBuilder	programové rozhraní PLC
Engineering Workplace	program sloužící pro nahrání projektu do ControlBuilderu
Filling	sekvence plnění
Functional Structure	funkční struktura
Hardware	jednotlivá zařízení, kterými jsou vybavena uvedená demo
Headmounted	druh provedení komunikačního protokolu ProfiBus
Interface	uživatelské rozhraní (ovládací panel)
IP adresa	adresa sloužící k síťovému propojení dvou či více zařízení
ModuleBus	optická sběrnice PLC
Off	tlačítko faceplatu sloužící ke spuštění sekvence vypouštění
On	tlačítko faceplatu sloužící ke spuštění sekvence plnění
PCC Display	displej zobrazující schéma regulační soustavy displeje, displeje zobr. stavy jednotlivých prvků
ProfiBus	průmyslová sběrnice
Reset	tlačítko faceplatu sloužící k resetu a inicializaci sekvence
Router	zařízení sloužící k bezdrátové komunikaci mezi počítačem a PLC
Sequences Demonstration Display	displej složený ze tří oblastí - Faceplatu, sekvenčního displeje a displeje zobrazujícího stavy jednotlivých prvků
SEQ Faceplate	navigační tlačítko pro spuštění sekvence procesu
Simulation On/Off	simulační tlačítka
Software	programové vybavení uvedených dem
Stop	tlačítko faceplatu sloužící k zastavení sekvence a zablokování všech přechodů v sekvenci
COMLI Slave	komunikační protokol sériového portu
Task	časování
Process Panel Builder	programové rozhraní pro ovládací panel (uživatelský interface)

1 Úvod

Cílem bakalářské práce je seznámit se s laboratorními modely (demy) od firmy ABB a to s jejich hardwarovými konfiguracemi, softwarovým vybavením a způsobem jejich programování. Zabývat se budu jak demem typu „800xA“, tak také i demem dávkovacího systému. V úvodní části charakterizuji v jednotlivých podkapitolách hardwarové prvky uvedených dem. Jelikož jsou obě dema vybavena téměř obdobnými komponentami, popíši podrobně jen demo „800xA“. U dema dávkovacího systému charakterizuji pouze ty prvky, které obsahuje navíc pro danou problematiku spojenou s regulováním výšky hladiny kapalin v nádržích.

Ve druhé části se budu věnovat podrobnějšímu popisu programovatelného logického automatu AC 800M, a to jeho hardwarovému vybavení a komunikaci s okolním zařízením. Zároveň si také představíme programovací rozhraní tohoto PLC.

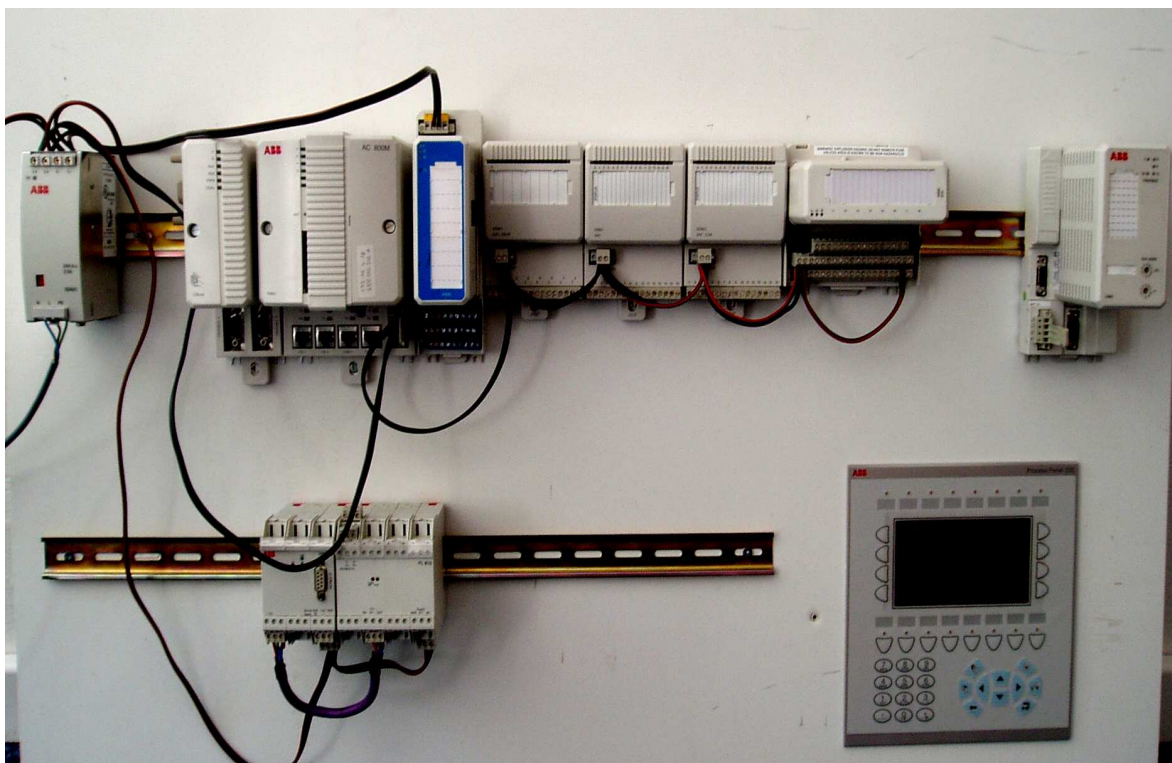
V dalším průběhu práce se už budu věnovat praktickému řešení problematiky spojené s komunikací dema 800xA (konkrétně mezi PLC a ovládacím panelem) a následnému ověření jeho funkčnosti prostřednictvím již vytvořené aplikace pro řízení kolejové dopravy.

V poslední části se pak budu snažit navrhnout a sepsat podrobné návody pro práci s oběmi modely uvedených dem a jejich spuštění tak, aby byly co nejjednodušší a mohl je uvést i obyčejný uživatel bez problémů do provozu.

2 Popis vybraných laboratorních modelů

2.1 Demo 800xA

Demo, které vidíme na obr. 1 slouží k řízení kolejové dopravy, popřípadě např. k řízení různých elektromotorů s použitím příslušného frekvenčního měniče.



Obr. 1 Schéma zapojení DEMA ABB800xA

Hardwarové vybavení uvedeného dema

Jelikož se jedná o modulární PLC, skládá se z více částí. Na hlavní procesorový modul se připojují další potřebné rozšiřovací moduly. Tyto mohou být digitální nebo analogové. Obsahují několik různých vstupů a výstupů. Existují i speciální moduly pro připojení různých měřicích zařízení jako např. teploměru PT100.

2.1.1 Procesorový modul AC 800M (typ PM860)

Procesorová jednotka (viz obr. 2) řídí prostřednictvím komunikačních a vstupně výstupních jednotek činnost celého systému. Umožňuje přímé připojení 192 I/O signálů přes elektrickou sběrnici nebo 1344 I/O signálů přes optickou. Tuto sběrnici výrobce nazývá ModuleBus. Komunikaci s okolím zajišťují porty CN1 – CN4 [MAREK, HABERNAL 2006].



Obr. 2 AC 800M – procesorová jednotka + komunikační jednotka

Technické údaje

- 2MB flash PROM (paměť pro firmware)
- 8MB SDRAM
- mikroprocesor MPC860 48MHz
- spotřeba 5W
- minimální napětí pro logickou 1 je 15V
- maximální napětí pro logickou 0 je 8V

2.1.2 Modul CI854A

Modul tohoto typu (viz obr. 3) umožňuje procesorové jednotce komunikovat s ostatními zařízeními (ať už další PLC nebo jen moduly pro rozšíření počtu I/O) po sběrnici Profibus. Tato průmyslová sběrnice vychází ze standardu sběrnice RS-485. Má sběrnicovou topologii [MAREK, HABERNAL 2006].



Obr. 3 Modul rozšíření komunikace CI854A

2.1.3 Vstupní/výstupní analogový modul AO801

Tento modul (viz obr. 4) generuje až 8 výstupních analogových signálů. Tyto signály jsou proudové v rozmezí $0 \div 20 \text{ mA}$ nebo $4 \div 20 \text{ mA}$. Z hlediska sledování chyb je výhodnější použít druhý ze zmiňovaných rozsahů. [MAREK, HABERNAL 2006]



Obr. 4 Modul analogových výstupů AO801

2.1.4 Modul diskretních vstupů V/V DI801

Obsahuje 16 diskretních vstupních kanálů. Ovládací napětí je 24V DC. Každý kanál má ochranu proudovým omezením, ochranu proti elektromagnetickému rušení, LED indikující stav vstupu a je opticky galvanicky oddělen od sběrnice ModuleBus (viz obr. 5). [MAREK, HABERNAL 2006]



Obr. 5 Modul diskretních vstupů DI801

2.1.5 Výstupní modul DO801

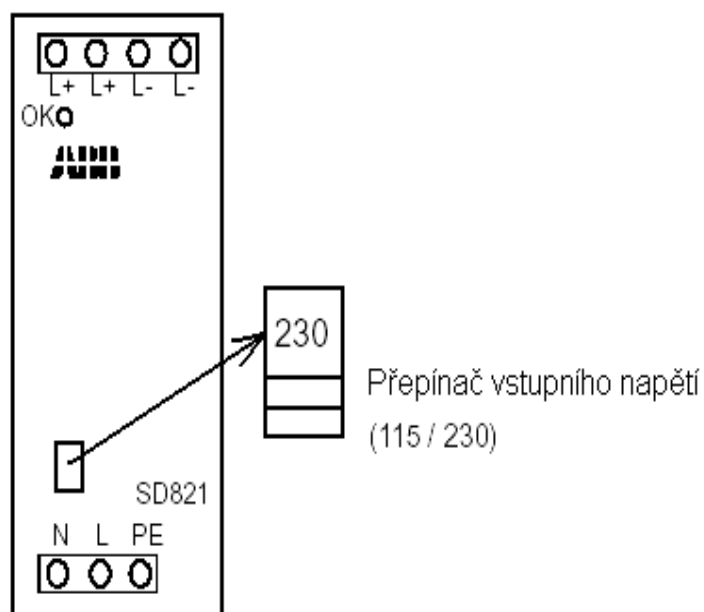
Modul (viz obr. 6) obsahuje 16 diskretních výstupních kanálů, na kterých se při sepnutí objeví napětí 24 V DC a maximální odebíraný proud je 0,5 A. Každý výstup je chráněn proti spojení nakrátko, přepětí, elektromagnetickému rušení, tepelnému přetížení a je opticky galvanicky oddělen od sběrnice ModuleBus a vybaven indikační LED. [MAREK, HABERNAL 2006]



Obr. 6 Modul diskretních výstupů DO801

2.1.6 Napájecí zdroj SD821

Dodává energii jak pro napájení samotných jednotek PLC, tak i pro výstupy výstupních modulů. Schéma zdroje je podrobně popsáno na obr. 7. Technické údaje jsou uvedeny v tab. 1. Napětí + 24 V je vyvedeno na svorkách označených L+ a 0 V je na svorkách L-. [MAREK, HABERNAL 2006]



Obr. 7 Napájecí zdroj SD821

Tab. 1 Parametry napájecího zdroje

Parametr	Hodnota
Jmenovitý výstupní proud	2,5 A
Jmenovitý výstupní výkon	60 W
Jmenovité výstupní napětí	24 V
Jmenovitý vstupní výkon	160 VA,
Dovolené kolísání vstupního	85 % -
Kolísání výstupního napětí	±2 %
Maximální výstupní proud	5 A

2.1.7 Frekvenční měnič ACS800-01/U1

ACS800-01/U1 (viz obr. 8) je měnič s montáží na stěnu, určený pro řízení střídavých motorů. Řízení motoru vychází z metody přímé regulace točivého momentu (DTC). Pro regulaci jsou měřeny a použity proudy ve dvou fázích a napětí ve stejnosměrném vedení. Proud ve třetí fázi se měří kvůli ochraně proti zemnímu spojení.



Obr. 8 Frekvenční měnič ACS800-01/U1

Technické údaje

Rozsah napětí	2 = 200/208/220/ 230 /240 V stř.
	3 = 380/ 400 /415 V stř.
	5 = 380/400/415/440/460/480/ 500 V stř.
	7 = 525/575/600/ 690 V stř.
Stupeň ochrany	B056 = IP 55 / UL typ 12
Odporové brzdění D150	= brzdový střídač
Filtr	E200 = filtr EMC/RFI pro druhé prostředí v soustavě TN
	E202 = filtr EMC/RFI pro první prostředí v soustavě TN
Ovládací panel	J405

Informace o ovládacím panelu

Displej typu LCD (viz obr. 9) má 4 řádky po 20 znacích. Jazyk se zvolí při spouštění.

Ovládací panel má čtyři provozní režimy:

- Režim zobrazení aktuálního signálu (klávesa ACT)
- Parametrický režim (klávesa PAR)
- Režim funkcí (klávesa FUNC)
- Režim výběru pohonu (klávesa DRIVE)

Použití kláves s jednoduchou šipkou, kláves se zdvojenou šipkou a klávesy ENTER závisí na operačním režimu panelu. Funkce přiřazené jednotlivým číslovaným klávesám jsou popsány v tab. 2.



Obr. 9 Ovládací panel frekvenčního měniče

Tab. 2 Funkce kláves na ovládání pohonu

Čís.	Užití
1	Start
2	Stop
3	Aktivace referenčního řízení
4	Směr otáčení VPŘED (VPRAVO)
5	Opačný směr otáčení (reverzace – (VLEVO))
6	Resetování poruchy
7	Změna mezi místním/dálkovým (externím) ovládáním

2.1.8 Ovládací panel PLC PP220

Tento ovládací panel, jinak také zvaný uživatelský interface (viz obr. 10), slouží k vyvolání samotného programu z PLC bez pomoci počítače. Může se jím také měnit programové nastavení či samotné PLC jednoduše naprogramovat. V praxi se velmi často využívá pro svou jednoduchou ovladatelnost a snadnou komunikaci mezi samotným PLC a obsluhou. Je to tzv. primární komunikační složka celého systému.

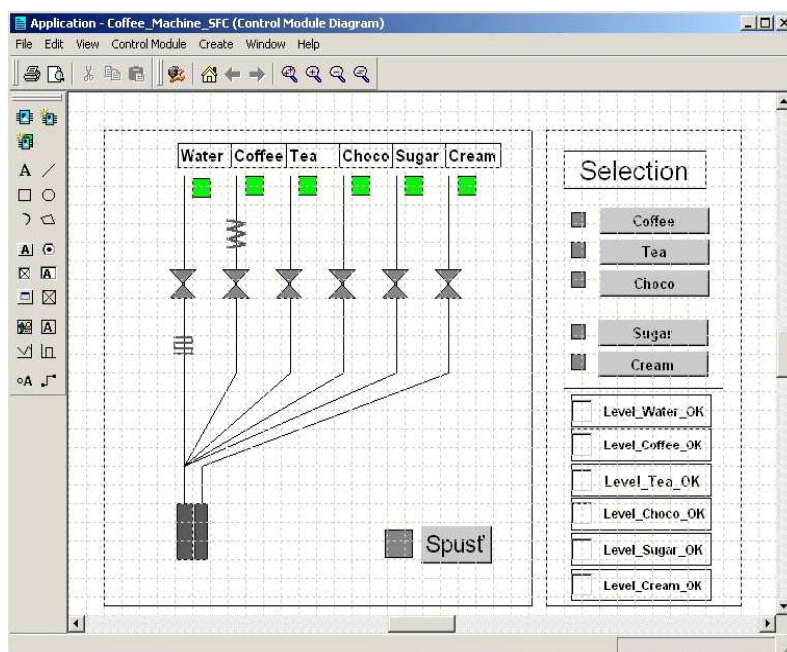
Panel je vybaven světelným monochromatickým LCD displejem s rozlišením 240x128 pixelů. Toto rozlišení umožňuje zobrazovat na velmi dobré úrovni jak textové, tak grafické údaje. Pod LCD displejem se nachází alfanumerická klávesnice, která čítá 16 programovatelných funkčních kláves. Dále je panel vybaven integrovanou 400 kb pamětí typu flash. Tuto lze pomocí flash karty rozšířit až na plných 8 MB. Co se týká komunikačního rozhraní je panel ve své dolní části vybaven dvěma porty typu RS 232C a RS 422. Uživatelský interface můžeme také vybavit modemem pro komunikaci přes Profibus nebo Ethernet. Softwarové vybavení panelu podporuje navíc použití až 10ti světových jazyků, které se mohou libovolně přepínat v průběhu dané aplikace. Uživatelský panel navíc umožňuje nahrání nové aplikace pomocí již zmíněné flash karty. To lze považovat za velkou výhodu hlavně v místech, kde nemáme k dispozici PC.



Obr. 10 Ovládací panel PLC

Programování aplikací pro ovládací panel

Pro možnost zapojení uživatelského interface je pro něj nutné nějakým způsobem danou aplikaci, kterou jím poté budeme následně ovládat naprogramovat. Vytváření jednotlivých grafických obrazovek je možné přímo v programovacím prostředí, kde je na tuto práci jednoduchý editor (viz obr. 11).



Obr. 11 Grafické prostředí návrhu obrazovek interface

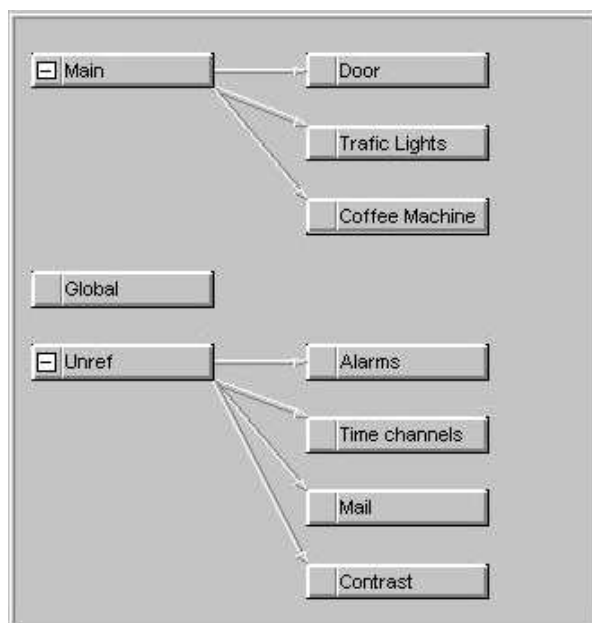
Po otevření se ukáže kreslicí plocha, která reprezentuje plochu, která se bude ukazovat na grafickém interface. Tuto plochu zleva obklopují možné prvky, které se dají tažením přemístit na kreslicí plochu. Nacházejí se zde jak pouze vizuální prvky jako například čára, text, obdélník, prvek apod. Nacházejí se zde také ovládací prvky pro interaktivní ovládání běžícího programu pomocí uživatelského interface jako je například tlačítko, zaškrtnutí apod.

Nad kreslicí plochou se klasicky zobrazuje uživatelské menu a pod ním se nachází zrychlená volba předem nastavených operací jako například vytisknout, náhled, editační předvolby práce se systémem a na závěr tlačítka pro zvětšování a zmenšování kreslicí plochy.

Tvorba grafické vizualizace ovládacího panelu

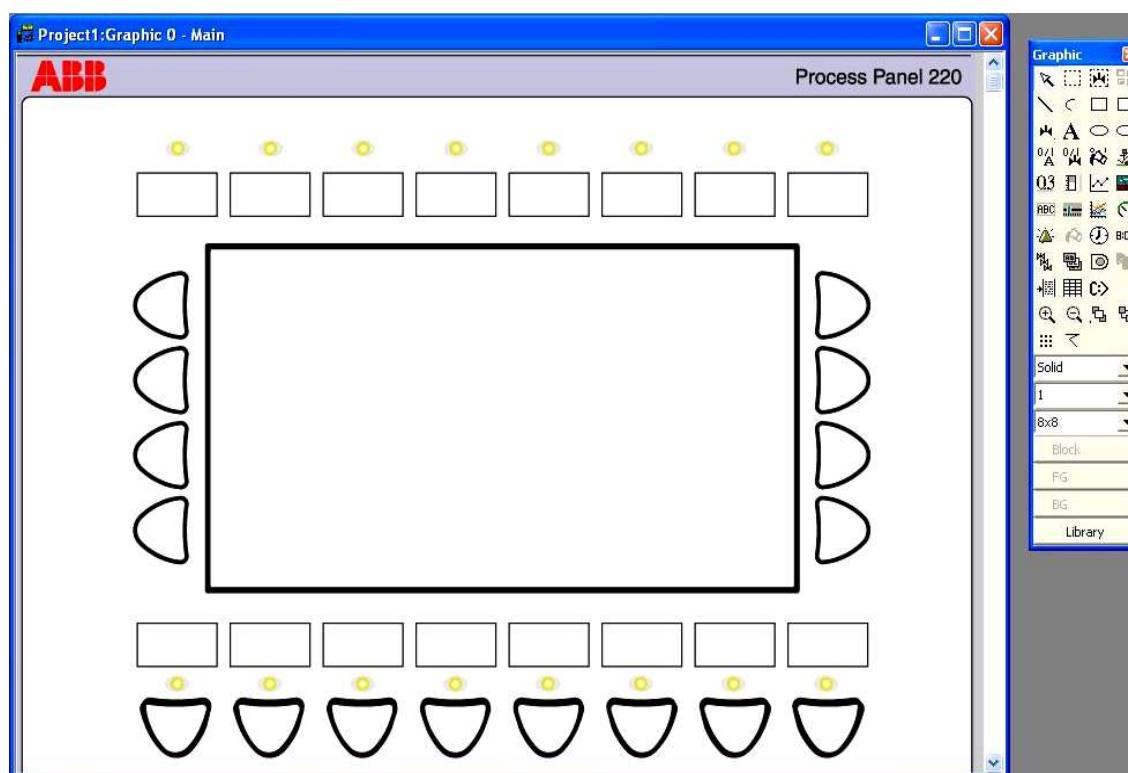
Kromě naprogramované aplikace popsané v předchozí kapitole je nutno vytvořit konkrétní grafické a textové prostředí pro uživatelský panel, se kterým budeme následně pracovat a ovládat jeho prostřednictvím danou aplikaci.

Tuto tzv. základní vizualizaci lze vytvořit v softwarovém rozhraní Process Panel Builder. Po jeho otevření a zvolení tvorby nového projektu se nám v hlavní obrazovce zobrazí okno se základní blokovou strukturou pro programování naší vizualizace (viz obr. 12). Jednotlivé textové bloky prezentují hlavní tlačítka klávesnice panelu, pod kterými budeme příslušnou aplikaci tvořit.



Obr. 12 Základní bloková struktura vizualizace

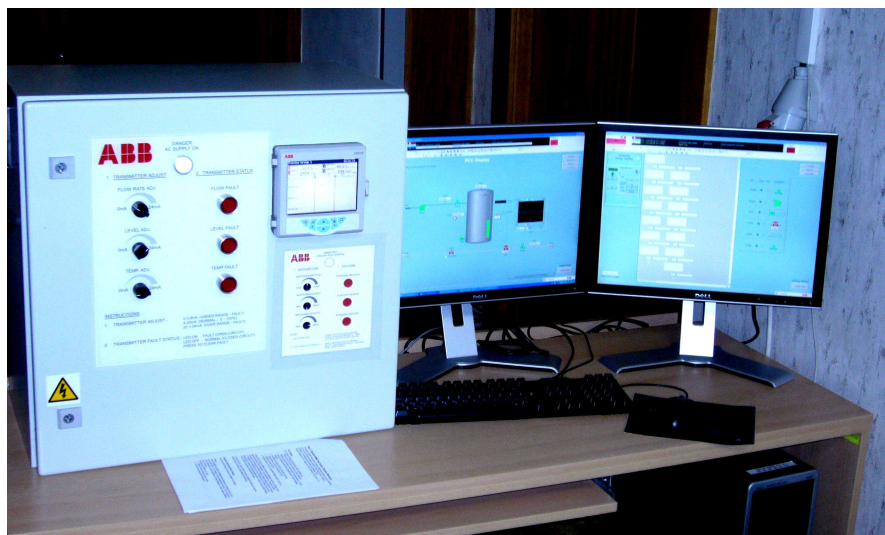
Po zvolení jakéhokoli z nich se nám objeví grafický editor prezentující LCD displej a klávesnici uživatelského interface (viz. obr. 13). Vedle něj máme navíc k dispozici panel nástrojů (viz obr. 13) pro vkládání různých grafických a textových polí do LCD displeje a možnost měnit libovolně jejich nastavení (velikost písma, tloušťka čáry apod).



Obr. 13 Grafický editor LCD displeje + panel nástrojů

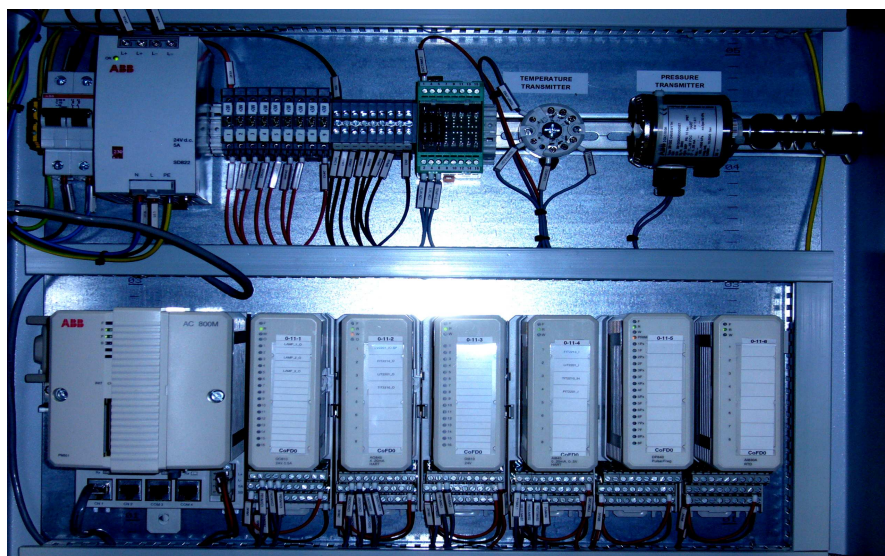
2.2 Demo dávkovacího systému

Demo, které vidíme na obrázcích (viz obr. 14 a obr. 15) se používá k regulování výšky hladiny kapalin v nádržích. Pro prezentování dané problematiky nám slouží software „Demo Dual Screen Operator Workplace“, ve kterém je realizována příslušná simulace vytvořená v softwarovém rozhraní Control Web, jíž můžeme vidět v jednotlivých monitorech na obr. 14. Tento model má tedy dvou obrazovkové operátorské pracoviště.



Obr. 14 Demo dávkovacího systému

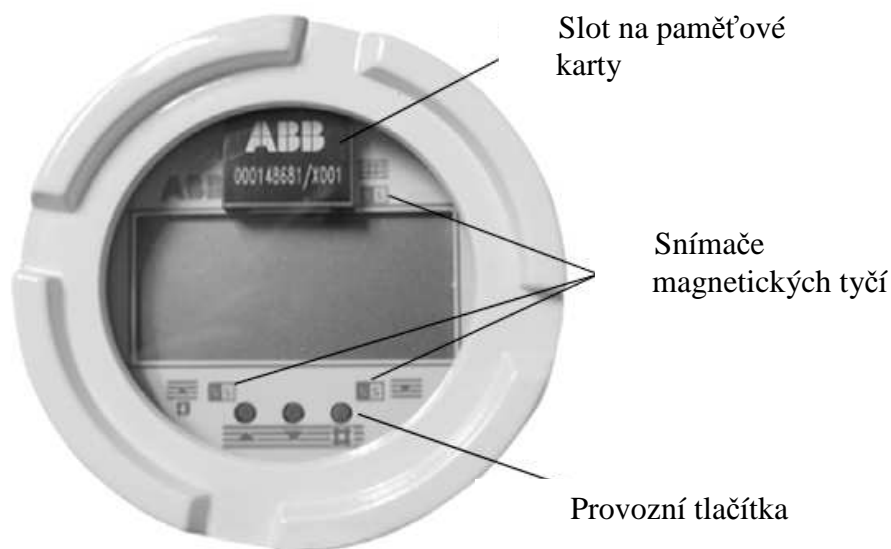
Po hardwarové stránce obsahuje toto demo kromě obdobných přídavných modulů i stejný typ PLC jako demo 800xA (viz obr. 15). Rozdíl je akorát v možnostech jeho programování. Lze jej programovat pouze prostřednictvím PC. Uživatelský interface zde není k dispozici.



Obr. 15 Schéma zapojení DEMA dávkovacího systému

2.2.1 Elektromagnetický průtokoměr FXE4000, PROFIBUS DP

Elektromagnetický průtokoměr (viz obr. 16) se používá k přesnému měření průtoku kapalin, které mají elektrickou vodivost větší než $5 \mu\text{S/cm}$ ($20 \mu\text{S/cm}$ pro demineralizovanou vodu).



Obr. 16 Displej průtokoměru

Technické údaje

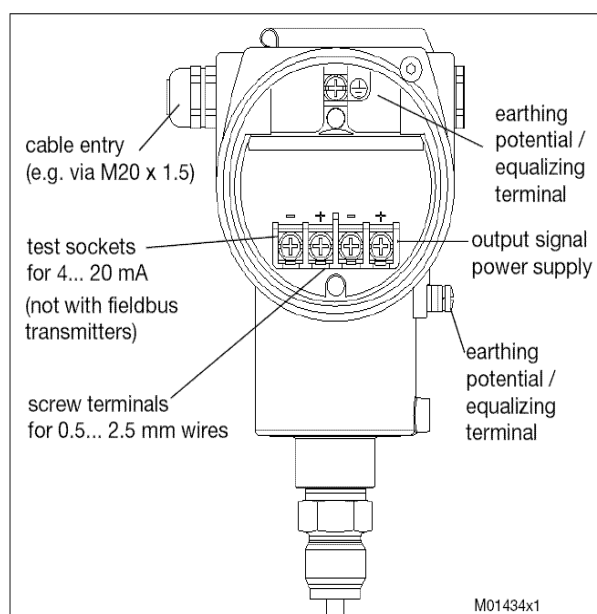
Napájení	Střídavý proud vysokého napětí: 100 - 230 V (-5/+10%)
Profibus DP protokol	Přenosová rychlost 9.6 k 1500 kbit/s
Displej	2x16- grafický LCD-dotykový maticový displej
Rozsah toku	nepřetržitě nastavitelný v rozsahu mezi 0.5 a 10 m/s
Správnost	$\leq 0.5 \%$ míry
Reprodukovatelnost	$\leq 0.15 \%$ míry

2.2.2 Tlakový senzor 265 GS, Hart

Tlakový senzor (viz obr. 17) měří absolutní tlak plynů, par a kapalin. Může měřit i teplotu kapaliny je-li v něm instalováno vnitřní tepelné čidlo. Schéma tlakového senzoru je podrobně popsáno na obr. 18.



Obr. 17 Tlakový senzor 265 GS, Hart



Obr. 18 Schéma tlakového senzoru

Technické údaje

Napájení	DC: 10,5 - 45 V
Hart protokol	protokol založený na sběrníčovém standardu 202 FSK
Displej	2-řádkový, 19-segment části alfanumerický displej
Výstupní signál	4 - 20
Adjusted	0 - 500 000 mmH ₂ O

2.2.3 Teplotní senzor TF 12 / TF 12-Ex

Tento senzor (viz obr. 19) slouží pro měření teploty a skládá se z teplotního článku Pt 100 (RTD) a A/D převodníku, který komunikuje po protokolu Profibus PA a je v provedení head mounted. Parametry senzoru lze nastavovat v počítačovém programu DSV401 přes ovládací rozhraní FDT/DTM TFx12-PA.



Obr. 19 Teplotní senzor TF 12 / TF 12-Ex [ABB 2005]

Technické údaje

Napájení	$U_s = 9...32 \text{ V DC}$
Baud rate Profibus PA	31.25 kbit/s
Jmenovitý odběr proudu	11.8 mA
Měřicí proud	200 μA
Odchylky měření Pt100	typ. < 0.2 % min. < 0.2 K min. < 20 μV
Měřicí rozsah	-200 až 850 °C

2.3 Softwarové vybavení uvedených dem

Program PLC lze vytvořit různými způsoby programování. Možnosti použít různých způsobů programování rozhoduje i v konečném výběru konkrétního PLC. Pro snadnější přechod mezi různými PLC systémy různých firem výrobci dodržují několik standardů. Je to možnost programovat ve třech základních módech: liniové schémata, blokové schémata

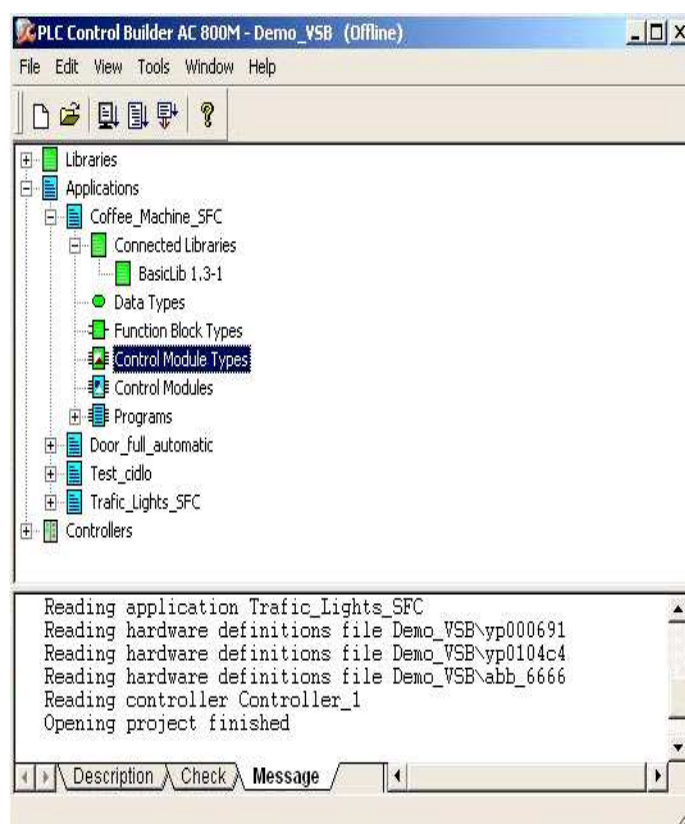
a přímo programování instrukcí. Dále je možno některé PLC naprogramovat pomocí vyšších programovacích jazyků. Nejvíce převažuje programovací jazyk C. Firma ABB (od které je i testované PLC) převážně volí cestu blokových schémat s implementací jednoduchého jazyka C. Softwarové rozhraní, pro které je programování tohoto PLC určeno se nazývá „Control Builder“.

Popis programového rozhraní Control Builder

Hlavní okno (viz obr. 20), které se objeví po spuštění programovacího prostředí na PC, se skládá z několika částí. Klasické menu v záhlaví nabízí standardní možnosti, jako je například „File“ pro nahrávání, ukládání, zavírání, zakládání projektu, nebo „Edit“. Pod ním se nachází pás s ikonami pro rychlejší spouštění vybraných operací.

Střední část tohoto okna obsahuje stromovou strukturu vytvářeného projektu. Najdeme zde pomocné knihovny, nastavení kontroléru, a soubory, týkající se vývojové aplikace. Nacházejí se zde jak vizuální nastavení obrazovek interface, tak i kód vykonávaného programu.

Pod touto částí se nachází pole, kde se zobrazují právě probíhané události, popisy událostí a kontrolní události.



Obr. 20 Obrazovka projektu

3 Popis PLC AC 800M a způsob jeho programování

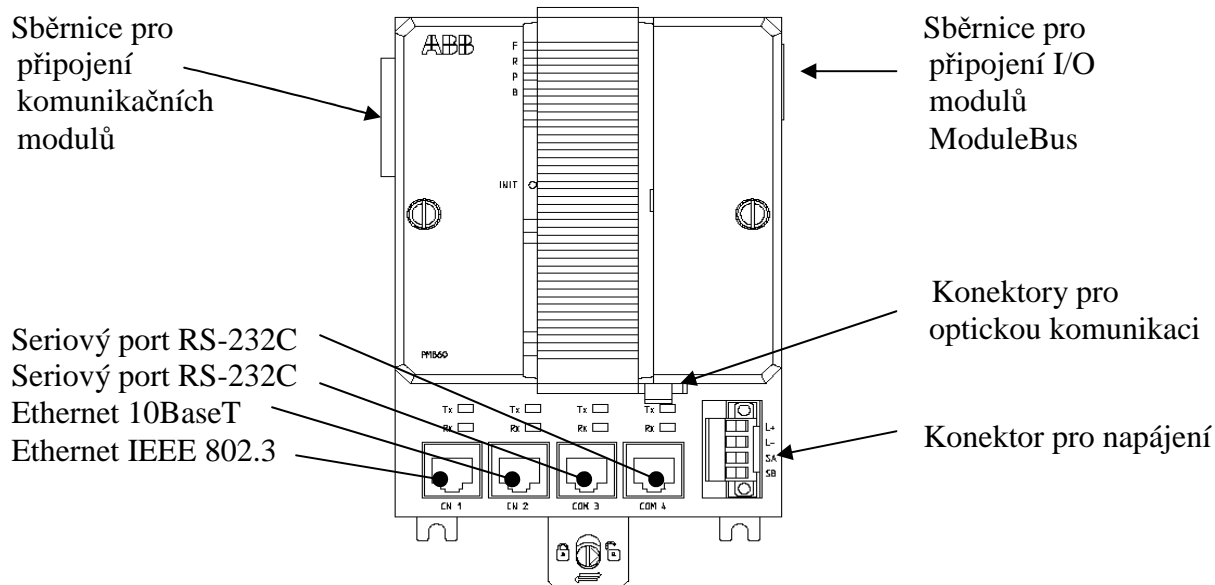
3.1 Programovatelný logický automat AC 800M

Samotné PLC (viz obr. 21) tvoří “srdce” v komunikaci s okolním prostředím. Tímto okolním prostředím máme na mysli především komunikační jednotku, díky které předává CPU informace a požadavky dalším příslušným jednotkám, jako jsou např. různé napájecí přídavné moduly (pro připojení vstupních a výstupních signálů (podrobněji popsány v druhé kapitole), či jiné doplňky. Příkladné zapojení PLC v praxi lze vidět na obr. 23. CPU tohoto modelu pracuje na frekvenci 48MHz. V PLC je obsažena integrovaná paměť o velikosti 8MB. Dále je tento model bohužel omezen na obsluhu pouze **jedné** nahrané aplikace.

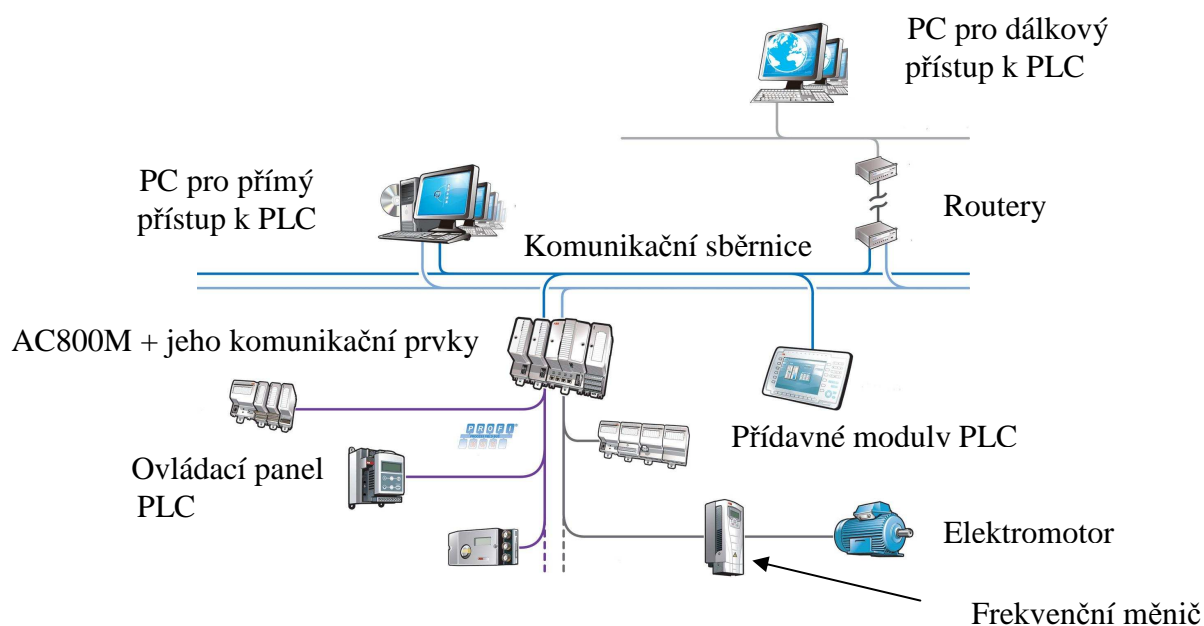


Obr. 21 PLC AC 800M(PM 860)

Modul PLC je vybaven dvěma ethernetovými porty, přes které s ním můžeme komunikovat prostřednictvím PC a nahrávat do něj příslušný software. Toto je ale možné i jinými způsoby. PLC má navíc ve své výbavě dva porty typu RS-232C, které mohou být použity pro tzv. dvoubodovou komunikaci s programovacími a ladícími prostředky pocházejících z různých jiných systémů či zařízení. Opomenout také nesmíme slot pro paměťové karty, který je umístěn na čelní straně panelu. Díky tohoto paměťového media se mohou rovněž nahrávat programy a data přímo do CPU. Modul PLC se svým příslušným komunikačním rozhraním je popsán podrobně na obr. 22.



Obr. 22 Detailní schéma procesorové jednotky PM860



Obr. 23 Komunikační schéma PLC

3.2 Základní programovací struktura

Program je možné zadávat do bloku „Programs“. Programuje se tedy pomocí blokových schémat a jednoduchého programovacího jazyka C.

Při otevření oka zdrojového kódu se otevře pomocné okno, které je rozděleno na několik základních částí. Nechybí samozřejmě uživatelské menu a lišty ikon pro rychlé spouštění vybraných událostí.

Dále je toto okno rozděleno do tří základních částí (viz Obr. 24). Nahoře se nachází okno s použitými proměnnými a schémata blokových struktur pro hlavní program. V tabulce proměnných je velice efektivní mít při celém vývoji konkrétního programového modulu tyto údaje vždy před sebou. Uspadňuje to samotnou práci s programovým modulem.

Zbytek okna je rozdělen do dvou vedle sebe položených částí. První (levá) část ukazuje hlavní blokovou strukturu programu. Lze z něj vyčíst nejenom významy jednotlivých bloků, ale i posloupnost. Hlavní program se vykonává cyklicky a při jednom cyklu se vykonají všechny bloky hlavního programu. Při programování je nutné na tuto vlastnost brát zřetel.

Pravá část nám ukazuje rozepsaný kód vybraného bloku. Tento kód je napsán v přizpůsobeném programovacím jazyce C. Přizpůsobeném proto, že pro využití všech možností PLC není potřeba všech možností programovacího jazyka C z počítačů PC. A naopak pro některé možnosti PLC není nutné vytvářet složité algoritmus obsluhy, ale je možné použít jednodušší nestandardní konstrukci příkazu.

Tento editor se také používá na programování uživatelského „interface“ (viz obr. 11).

The screenshot displays the 'Program - Coffee_Machine_SFC_Program_pro_gnael' window. It features a menu bar (Editor, Edit, View, Insert, Tools, Window, Help) and a toolbar. Below the toolbar is a table of variables:

Name	Data Type	Attributes	Initial Value	I/O Address	Access Variables	Description
1 PTCounter	dint		10			
2 PTGrind_Coffe	dint	constant	12			
3 PTWater	dint	constant	4			
4 PTValve	dint	constant	30			
5						
6						
7						

Below the table are tabs for 'Variables' and 'Function Blocks'. The main workspace is divided into two panes. The left pane shows a block diagram with the following blocks and connections:

- Selection** block (P1 Selection_P1, N Selection_N) connects to **Check_Sput**.
- Preparation** block (P1 Preparation_P1, N Preparation_N) connects to **Check_Preparation**.
- WaterOpen** block (P1 WaterOpen_P1, N WaterOpen_N) connects to **Check_Water**.
- Mixing** block (P1 Mixing_P1, N Mixing_N) connects to **Check_Mixing**.
- Finishing** block (P1 Finishing_P1, N Finishing_N, P0 Finishing_P0) connects to **Check_Finishing**.

The right pane shows the ladder logic code for the 'Selection' block:

```

if PLevel_Water_OK then
(* Select Choco *)
if PSel_Choco and PLevel_Choco_OK then
PInd_Choco := true;
PInd_Tea := false;
PInd_Coffee := false;
PSel_Tea := false;
PSel_Coffee := false;
PSel_Choco := false;
end_if;
(* Select Tea *)
if PSel_Tea and PLevel_Tea_OK then
PInd_Tea := true;
PInd_Choco := false;
PInd_Coffee := false;
PSel_Coffee := false;
PSel_Choco := false;
PSel_Tea := false;
end_if;
(* Select Coffee *)
if PSel_Coffee and PLevel_Coffee_OK then
PInd_Coffee := true;
PInd_Choco := false;
PInd_Tea := false;
PSel_Tea := false;
PSel_Choco := false;
PSel_Coffee := false;
end_if;
(* Select Sugar or Cream *)
if (PInd_Choco or PInd_Tea or PInd_Coffee) then
PSugar( Clk := PSel_Sugar );
if PSugar.Q and PLevel_Sugar_OK then
PInd_Sugar := not PInd_Sugar;
PSel_Sugar := false;
end_if;
PCream( Clk := PSel_Cream );
if PCream.Q and PLevel_Cream_OK then
PInd_Cream := not PInd_Cream;
PSel_Cream := false;
end_if;
else
PInd_Sugar := false;
PInd_Cream := false;
end_if;
end_if;

```

The bottom status bar shows the current row and column (Row 1, Col 1) and the time (14:35).

Obr. 24 Obrazovka programových algoritmů

4 Ověření funkčnosti demo 800xA

Odstranění komunikační poruchy mezi ovládacím panelem a PLC

Při ověřování celkové funkčnosti demo 800xA tzn. především, zda byla dobře nahrána příslušná aplikace do PLC a jeho správné propojení s okolním hardwarem, byla zjištěna komunikační závada na uživatelském interface. Na jeho LCD displeji se zobrazovalo chybové hlášení „Comm Error“.

Uživatelský panel je k PLC připojen prostřednictvím sériového kabelu do portu „COM3“. Nejprve se tedy zkontrolovalo toto propojení, které bylo ovšem v pořádku. Bylo tedy zřejmé, že daný problém spočívá v nahrané softwarové aplikaci buď v PLC nebo v uživatelském interface. K tomu, aby bylo možno řídit prostřednictvím ovládacího panelu jakoukoli aplikaci je nutno, aby tato byla nahrána jak v PLC, tak její zjednodušená verze (ve formě vizualizace) právě i v ovládacím panelu. A jelikož to byl právě uživatelský interface, který hlásil poruchu, dalo se předpokládat, že k němu nebude v PLC nahraný příslušný software nebo v něm bude chyba.

K ověření tohoto předpokladu byla k dispozici již vytvořená aplikace s názvem „Demo VŠB“ a k ní vytvořená příslušná vizualizace, tedy software pro uživatelský panel. Tyto bylo zapotřebí samozřejmě do PLC a do panelu nahrát. Co se týká PLC, tak pro vložení daného projektu byl použit již popisovaný software v kapitole 2.3 „Control Builder M“. Podrobný postup daného nahrávání aplikace bude popsán v následujících kapitolách, které se budou této problematice věnovat.

Když byl tedy projekt v PLC nahrán, nezbývalo nic jiného než provést stejný pracovní postup i u uživatelského interface. Zde se zjistilo, že se pro vkládání aplikací do panelu již nepoužívá stejný software jako u PLC, ale speciální programové rozhraní „Process Panel Builder“, které bylo vytvořeno pro obecnou správu panelu. Tím je na mysli především jeho programování.

Jelikož byl v danou chvíli uživatelský interface připojen k PLC, bylo nutno tento od něj odpojit a připojit jej k PC opět prostřednictvím sériového kabelu a to konkrétně do portu „COM1“! Následně se otevřela daná aplikace v „Process Panel Builderu“ a bylo spuštěno nahrávání do panelu. V průběhu nahrávání se automaticky přepisovala původní aplikace, která již byla v panelu dříve nahrána. To samé platilo mimochodem i v předchozím kroku u PLC.

Po dokončení daného procesu se panel odpojil od PC a byl připojen opět k PLC. Panel se následně po chvilce s PLC zinicizoval a chybové hlášení, které se vyskytovalo na jeho displeji „Comm Error“ zmizelo.

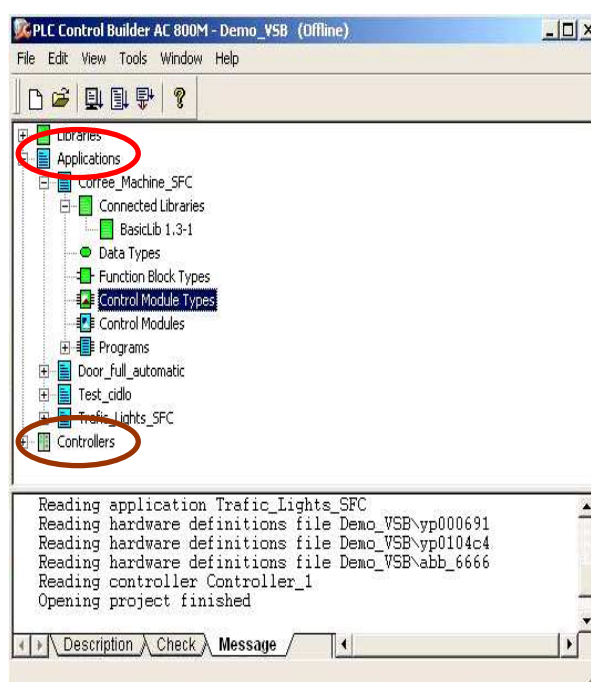
Zda byla skutečně komunikační závada úspěšně odstraněna se poté ověřilo přímo na panelu, kdy bylo možno v simulačním režimu s danou aplikací prostřednictvím PLC bez problémů pracovat.

Ověření funkčnosti prostřednictvím příslušné aplikace

Další praktická část této práce se věnovala odzkoušení funkce PLC na již předem vytvořené aplikaci, která měla za úkol řídit kolejovou dopravu vláček. Tato aplikace byla již v předchozích dobách využívána k prezentaci praktického využití PLC pro studenty předmětu „Základy automatizace“.

Úkolem práce bylo kromě simulování a odzkoušení její funkčnosti přidat tuto již ke zmíněné aplikaci „Demo VŠB“, která byla již použita pro odstranění komunikační závady v předchozím bodě.

K tomuto procesu se opět použilo již známé softwarové rozhraní „Control Builder“. Nejprve se v něm tedy otevřel projekt „Demo VŠB“. Pro přidání nové aplikace z jiného projektu se rozklikla ikonka „Applications“ (viz obr. 25) a byla zvolena položka „Insert application“. Následně se objevilo dialogové okno, kde se označil příslušný adresář s danou aplikací a ta se vložila mezi ostatní (ty které již byly v „Demu VŠB“ vytvořeny „defaultně“).

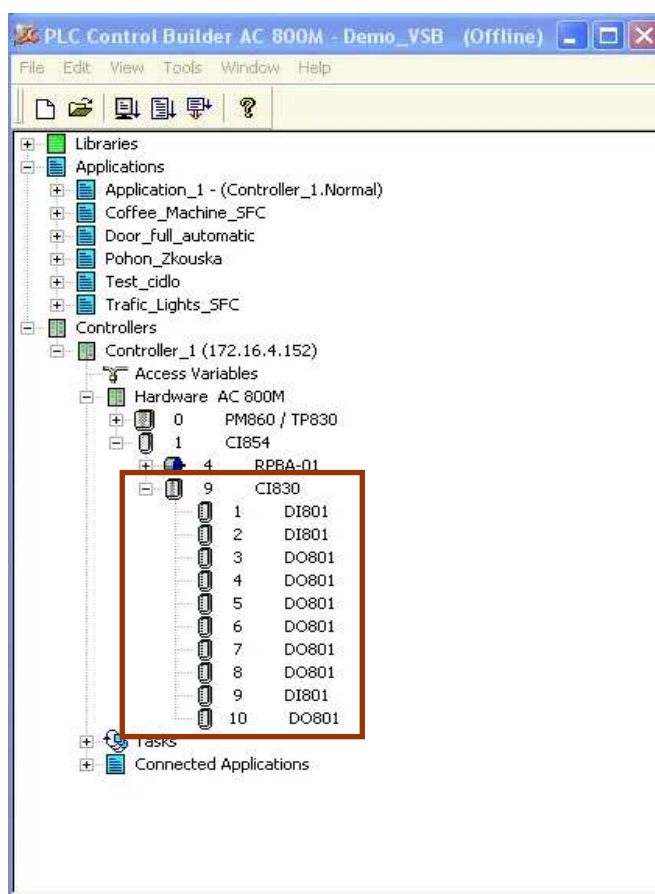


Obr. 25 Obrazovka projektu

Poté bylo ještě nutno přidat do „Control Builderu“ komunikaci příslušnou dané aplikaci pro řízení kolejové dopravy. Toto bylo učiněno obdobně jako u nahrávání nové aplikace a to tak, že se rozklikla ikonka „Controllers“ (viz obr. 25) a byla zvolena položka „Insert controller“. Znovu poté vyskočilo dialogové okno pro určení adresáře, kde se daný soubor obsahující komunikační nastavení nachází.

Po jeho nahrání do kontroleru byly již s původní („defaultní“) komunikací nastaveny tedy pro PLC komunikace dvě. Následně se zjistilo, že je to problém, neboť když byla celá aplikace později nahrávána do PLC, tak se objevilo chybové hlášení, že pro PLC může být nastavena pro jeden projekt pouze jedna komunikace a nikoli dvě, jak to bylo v tomto případě.

Jelikož „Control Builder“ neumožňuje zkopírovat příslušný komunikační hardware z jedné komunikace do druhé, musel se tento ručně všechen nastavit. Na obr. 26 jsou označeny jednotlivé prvky (moduly), které bylo nutno pro zprovoznění komunikace kolejové dopravy přidat.



Obr. 26 Moduly pro řízení kolejové dopravy

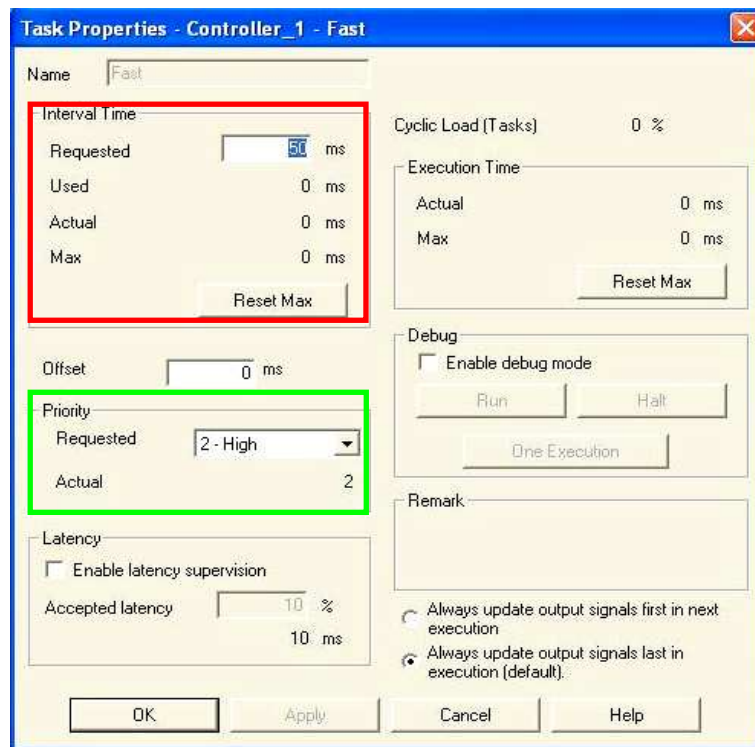
Kromě nainstalování jednotlivých modulů bylo nutno tyto také nějakým způsobem propojit s naprogramovanou aplikací. Tento proces se provedl tak, že se poklepáním na jednotlivé moduly dostalo do jejich nastavovacího rozhraní a u jejich jednotlivých vstupních adres byla nastavena příslušná aplikace a k ní program, který se bude na daném vstupu vykonávat (viz obr. 27).

Channel	Name	Type	Variable	I/O Description
IX1.9.1.1	Input 1	bool	Application_1.Program2.1_S1	
IX1.9.1.2	Input 2	bool	Application_1.Program2.1_S2	
IX1.9.1.3	Input 3	bool	Application_1.Program2.1_S3	
IX1.9.1.4	Input 4	bool	Application_1.Program2.1_S4	
IX1.9.1.5	Input 5	bool	Application_1.Program2.1_S5	
IX1.9.1.6	Input 6	bool	Application_1.Program2.1_S6	
IX1.9.1.7	Input 7	bool	Application_1.Program2.1_S7	
IX1.9.1.8	Input 8	bool	Application_1.Program2.1_S8	
IX1.9.1.9	Input 9	bool	Application_1.Program2.1_S9	
IX1.9.1.10	Input 10	bool	Application_1.Program2.1_S10	
IX1.9.1.11	Input 11	bool	Application_1.Program2.1_S11	
IX1.9.1.12	Input 12	bool	Application_1.Program2.1_S12	
IX1.9.1.13	Input 13	bool	Application_1.Program2.1_S13	
IX1.9.1.14	Input 14	bool	Application_1.Program2.1_S14	
IX1.9.1.15	Input 15	bool		
IX1.9.1.16	Input 16	bool		
IW1.9.1.17	All Inputs	dword		
IW1.9.1.18	Channel status	dword		
IW1.9.1.19	UnitStatus	dint		

Obr. 27 **Obrazovka nastavení výstupních adres jednotlivých modulů**

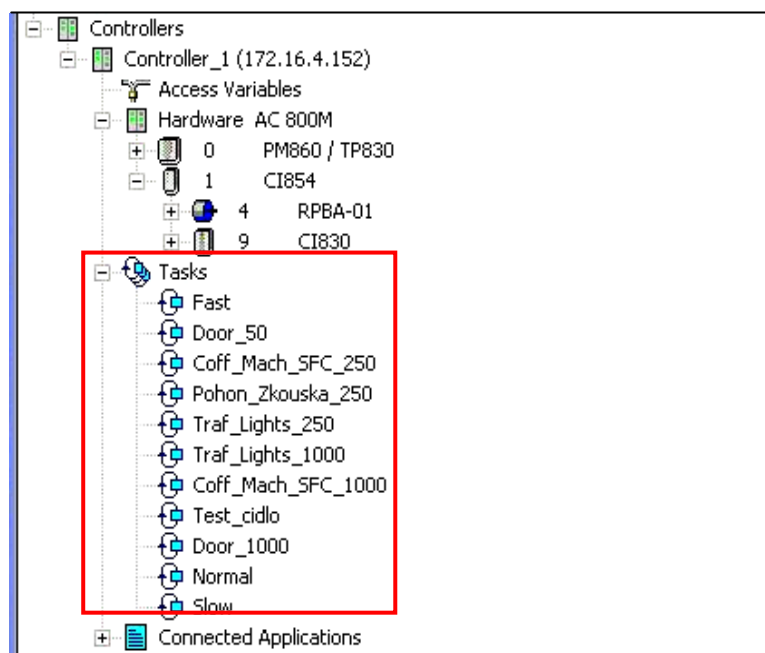
Po nastavení jednotlivých komunikačních adres každého modulu (viz obr. 26) bylo ještě v poslední fázi zapotřebí nastavit příslušné časování, tzv. „tasky“ u jednotlivých programů v aplikaci. Nejprve se tyto „tasky“ musely nadefinovat v komunikaci PLC. Vytvoření jednotlivých „tasků“ se provedlo obdobně jako tomu bylo u přidání nové aplikace nebo komunikace. V položce „Controllers“ se rozklikla ikonka s názvem „Tasks“ a byl zvolen „New task“. Následně se objevila obrazovka pro podrobné nastavení daného „tasku“ (viz obr. 28). Tato je rozdělena na několik oblastí. Pro nás jsou důležité ale pouze dvě z nich a to „Interval Time“ a „Priority“. Ostatní se zanechaly nastaveny „defaultně“.

U zmíněných oblastí se vyplnila pouze položka „Requested“. V oblasti „Interval Time“ tato položka slouží k nastavení časové prodlevy mezi jednotlivými operacemi v milisekundách (ms). V oblasti „Priority“ se zde určuje priorita, s jakou se bude nastavené časování v dané operaci využívat, např. „2-High“ (vysoká - viz obr. 28)



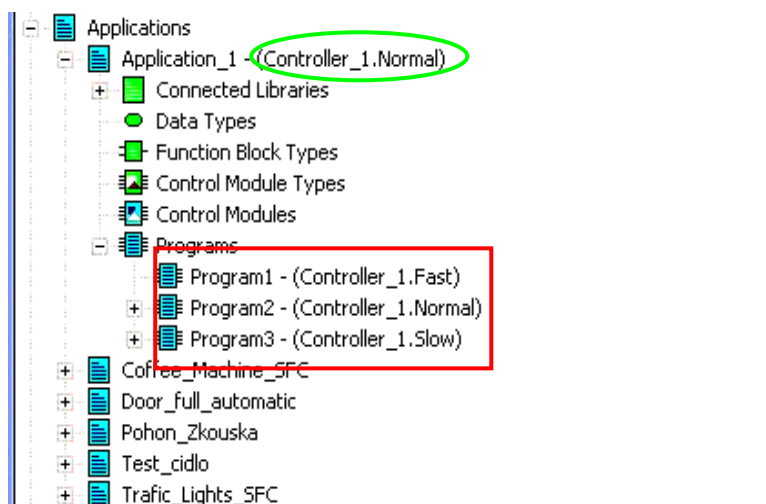
Obr. 28 Obrazovka nastavení časování (tasku)

Pro aplikaci řízení kolejové dopravy se nastavily „tasky“ tři a to rychlý („Fast“), normální („Normal“), pomalý („Slow“) s prodlevami 50ms, 100ms a 1000ms. a prioritami „2-High“, „3-Medium“ a „4-Low“. Jednotlivé „tasky“ pro danou aplikaci lze vidět na obr. 29.



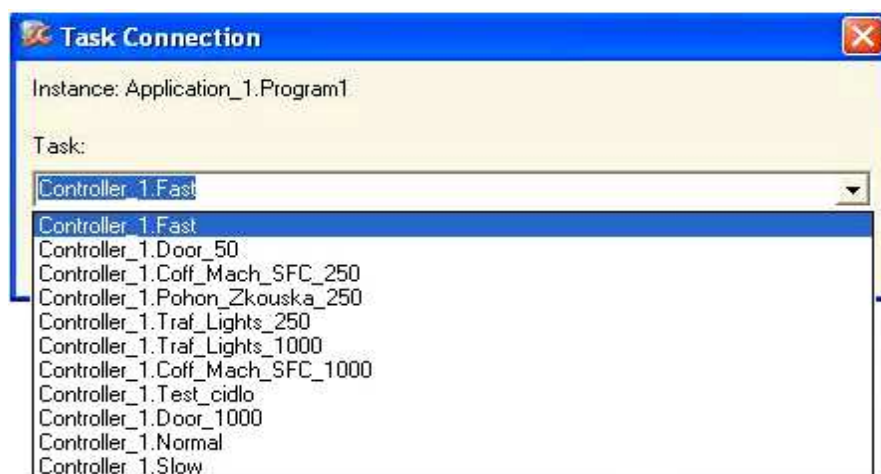
Obr. 29 „Tasky“ v komunikačním rozhraní PLC

Po naprogramování těchto „tasků“ je bylo tedy nakonec, jak už bylo zmíněno, nutno přidat k jednotlivým programům naší aplikace. Postupováno bylo tedy opět úplně stejně jako v předchozích případech. Nejprve se „rozbalila“ ikonka „Applications“, v ní poté ikonka „Application_1“ (aplikace pro řízení kolejové dopravy). a nakonec ikonka „Programs“ s jednotlivými programy (viz obr. 30).



Obr. 30 Přidávání „tasků“ do aplikace

Tady se postupně individuálně u každého z nich nastavily v nabídce „Task Connection“ jednotlivé vytvořené „tasky“ z komunikačního rozhraní, které je v případě této aplikace pod názvem „Controllers“ (viz obr. 31). Další a zároveň jako poslední věc před spuštěním celé aplikace bylo potřeba ještě navolit „task“ pro celkovou aplikaci („Application_1“) pod kterým bude „defaultně“ při jejím naběhnutí pracovat. Zde se postupovalo obdobně jako u jednotlivých programů a nastavil se zde „task“ „Normal“ (viz obr. 30)



Obr. 31 Výběr a nastavení jednotlivých tasků

Když už bylo vše hotovo nezbývalo nic jiného než celý projekt se všemi aplikacemi nahrát do PLC a ověřit jeho funkčnost. Konkrétně prostřednictvím zmíněného řízení kolejové dopravy.

K ověřování byl použit laboratorní model kolejiště, který je s PLC propojen prostřednictvím sběrnice „Profibus“. Přesněji řečeno propojen přímo s moduly vstupních a výstupních signálů, které jsou ke kolejišti namontovány. Jednotlivé z nich slouží např. pro řízení výhybek a semaforů na kolejové trati prostřednictvím speciálního ovládacího panelu. Tento se k daným modulům připojuje pomocí čtyř LPT (paralelních) kabelů, které jsou patřičně očíslovány tak, aby nedošlo k jejich chybnému propojení. Na panelu je potom obsaženo zmenšené schéma kolejové trati a jednotlivá tlačítka pro ovládání kolejové dopravy podle příslušně naprogramované aplikace.

Po nahrání projektu do PLC se moduly i s ovládacím panelem zinicizovali a bylo možno provést simulaci. Na kolejiště se nainstalovaly zmenšené modely vláčků a byl řízen jejich pohyb přes panel. Simulování probíhalo dobře a bez jakýchkoli chyb. Tudíž byla daná aplikace nastavena a nahrána do PLC úspěšně.

5 Návod pro práci s modelem dávkovacího systému

Přihlášení do systému

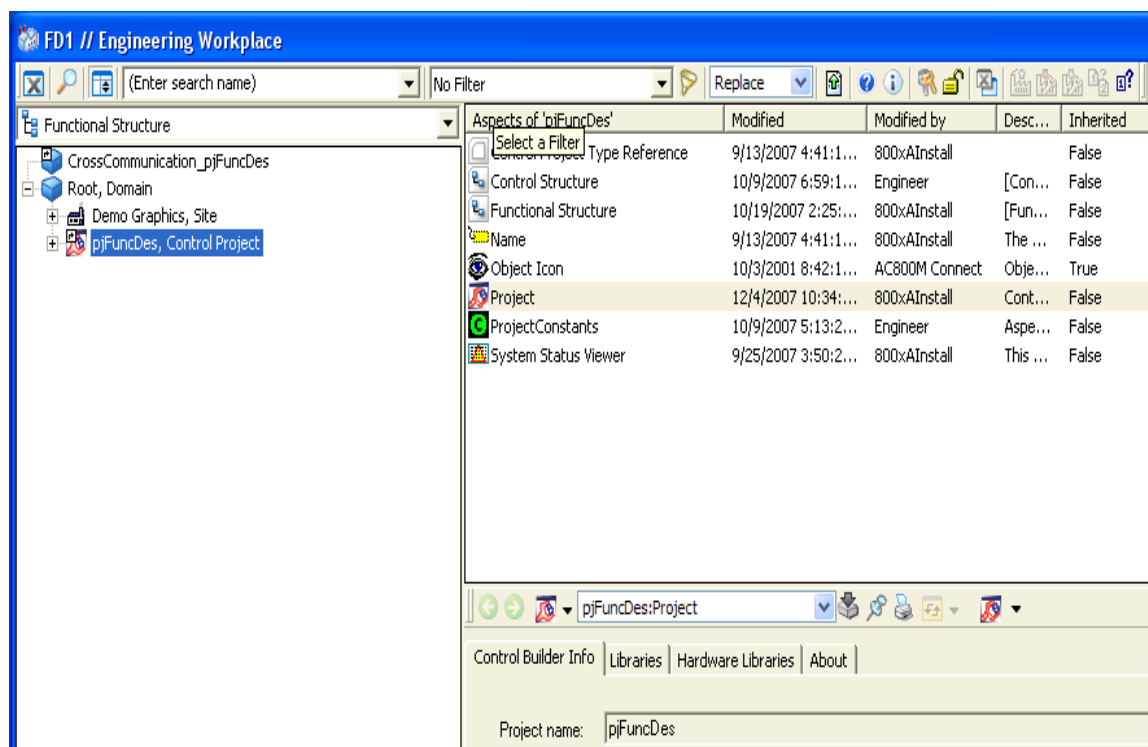
Po naběhnutí operačního systému počítače je nutno počkat, až se spustí všechny služby systému 800xA, což může trvat několik minut. Zelená ikonka v pravém dolním rohu plochy operačního systému na obr. 32 signalizuje, že je systém připraven k použití.



Obr. 32 Zelená ikonka uprostřed signalizuje připravenost systému

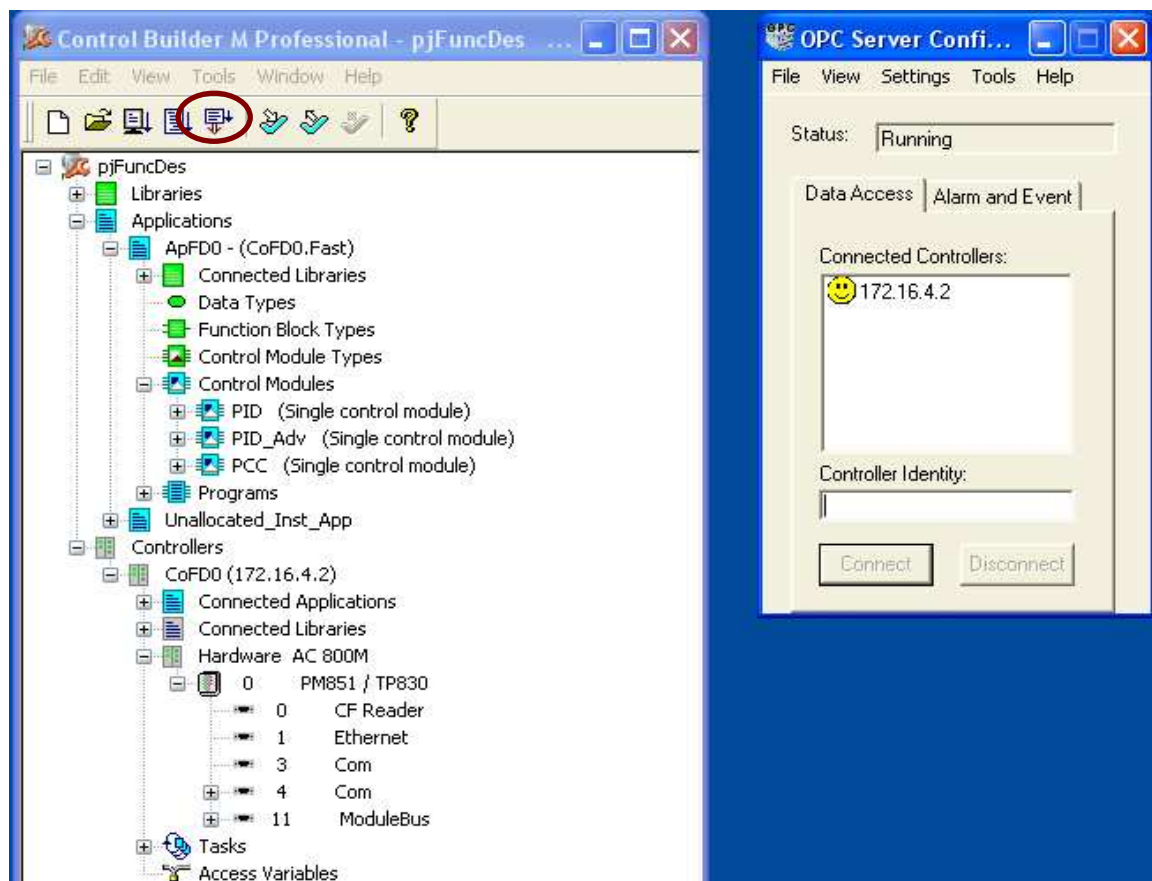
Spuštění a nahrání projektu do kontroleru

Pokud proběhlo vše z předcházejícího kroku v pořádku, je nutné nahrát a poté spustit náš projekt v kontroleru. Nejprve rozklikneme na ploše ikonku s názvem „ePlant“ a zvolíme program „Engineering Workplace“ (viz obr. 33). Poté se nám otevře okno, kde uvidíme funkční strukturu „Functional Structure“. Zde rozklikneme položku „Root“ a poté „pjFuncDes“, „Control Project“. Vedle v okně aspektů, které slouží k popisu objektu, v našem případě „Control Project“, zvolíme „Aspect Project“.



Obr. 33 Obrazovka programu Engineering Workplace

Projekt se nám následovně otevře v programu „Control Builder M“ (viz obr. 34), v němž byl vytvořen. Pro komunikaci mezi „Control Builderem M“ a „Plant Explorerer“ je nutné spustit „OPC server“ pro PLC AC 800M, který je umístěn na ploše.



Obr. 34 Obrazovka programu Control Builder M Profesional

Jestliže proběhlo vše správně, měli bychom vidět žlutého „smajlíka“ u IP adresy kontroleru (viz obr. 34). Pro nahrání aplikace do kontroleru je nutno zvolit pátou ikonku zleva v hlavním panelu.

Jako další nám vyskočí okno „Online analysis“, kde máme na výběr způsob nahrání. (volba „**Continue**“ s aktuálními hodnotami proměnných, nebo volba „**Cold Restart All**“, kdy budou hodnoty proměnných nastaveny na hodnotu uvedenou v „Control Builderu“ jako „**Initial Value**“). V okně zpráv „Control Builderu“ bychom pak měli vidět zprávu „You are now online!“

Otevření programového rozhraní DEMO Dual Screen Operator Workplace

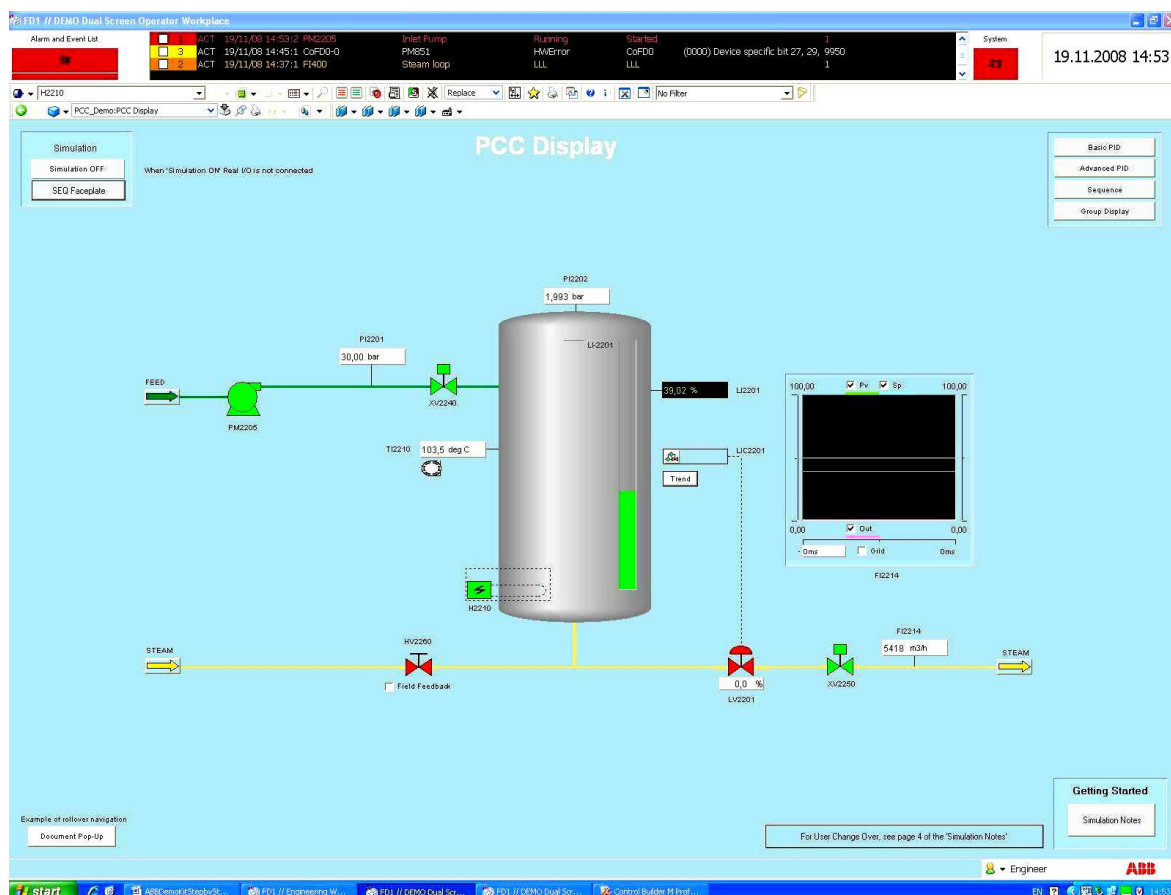
Pro otevření tohoto programu rozklikneme opět na ploše ikonu „ePlant“ a zvolíme „DEMO Dual Screen Operator Workplace“. Po jeho otevření se nám objeví dva displeje: „PCC Display“ na levém monitoru a „Sequences Demonstration Display“ na pravém monitoru.

Popis PCC displeje

Tento displej (viz obr. 35) zobrazuje jednoduchou nádrž a skládá se z přítokového ventilu (XV2240), přítokové pumpy (PM2205), odtokového ventilu (XV2250), ventilu řízeného PID regulátorem (LV2201), dále pak obsahuje manuální odtokový ventil, kde k otevření slouží “Field Feedback checkbox“, dvoutlakové snímače (PI2201 a PI2202) na přítoku, teplotní senzor (TI2210), ohříváč (H2210), průtokový snímač (FI2214), snímač hladiny (LI2201) a kontroler hladiny nádrže (LIC2201).

V levém horním rohu je ovládání simulace a navigační tlačítko pro displej řízení sekvence procesu (“SEQ Faceplate”). Nabízejí se zde dvě možnosti. Simulační tlačítko “Simulation On“ znamená, že reálné vstupy a výstupy nejsou připojené. Naopak v “Simulation Off“ připojené jsou a lze je ovládat přímo na „DEMO Kitu“ pomocí potenciometrů. Jedním se nastavuje teplota vody v nádrži, druhým výška hladiny v nádrži a třetím propustnost odtokového ventilu.

V pravém horním a levém dolním rohu jsou navigační tlačítka pro sekvenci procesu, skupinu displejů a odpovídající dokumentaci.



Obr. 35 Obrazovka PCC displeje

Sequence Demonstration Display

Displej je rozdělen do 3 oblastí:

Oblast A

Vlevo na displeji je umístěn „Faceplate“ řízení sekvence („Faceplate“, který může být vyvolán také z „PCC displeje“ pomocí navigačního tlačítka „SEQ Faceplate“). Tento „faceplate“ je použit operátorem k řízení sekvence a dává celkový přehled o stavu sekvence (viz obr. 36).

„Faceplate“ je rozdělen do **5 částí**: Nejvyšší část zobrazuje jméno a popis. Druhá část zobrazuje stav sekvence. Třetí část zobrazuje detaily jako současný krok, ve kterém se sekvence nachází, čas kroku, atd. Čtvrtá část zobrazuje různé příkazové tlačítka dostupné pro řízení sekvence.

Tlačítko pro Auto / Manuální řízení

“On” tlačítko ke spuštění sekvence plnění

“Off” tlačítko ke spuštění sekvence vypouštění

“Stop” tlačítko k zastavení sekvence a zablokování všech přechodů v sekvenci

“Reset” tlačítko k „resetu“ a inicializaci sekvence

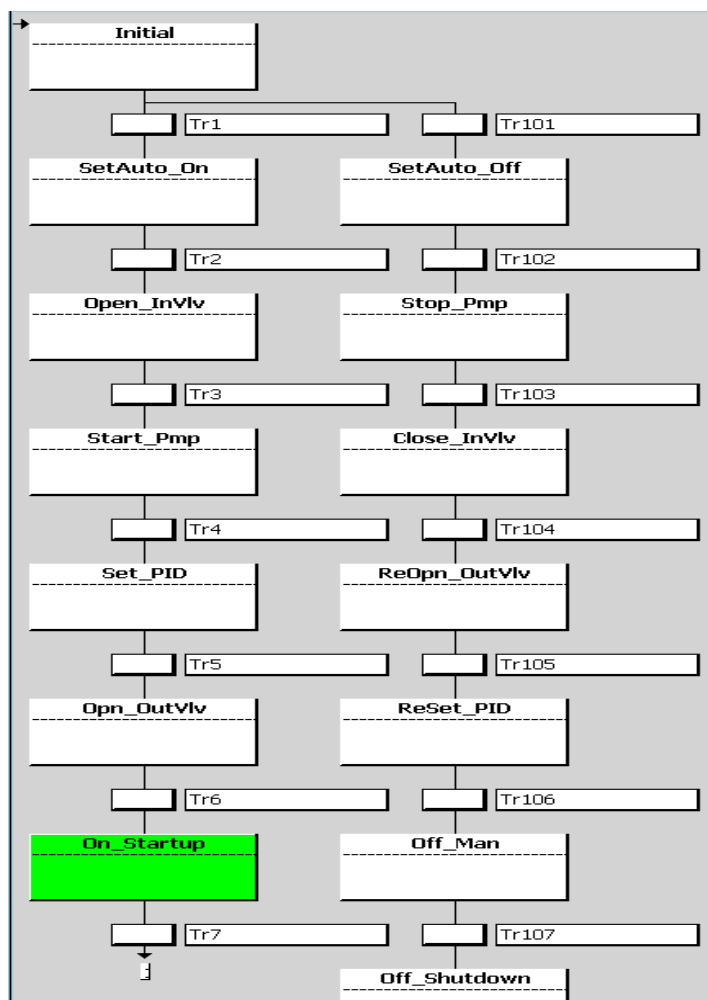
Rozšířený „faceplate“ zobrazuje „error“, „interpoly“, stavové a příkazové stránky.



Obr. 36 Obrazovka „Faceplate“

Oblast B

Vedle „faceplatu“ je hlavní sekvenční displej (viz obr. 37). Lze v něm vidět stav, ve kterém se sekvence nachází. V pravém sloupci je sekvence plnění („Filling“) a ve druhém je vypouštění („Draining“).




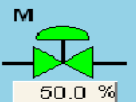




Obr. 37 Sekvenční displej

Oblast C

Displej zobrazující stavy jednotlivých prvků (pump, ventilů a kontrolérů) v procesu (viz obr. 38). Stav „**Auto**“ znázorňuje stav objektu a to buď „Auto“ nebo „Manual“. Stav „**PCC**“ znázorňuje buď aktivní nebo „interpol“ stav na daném objektu.

Každý objekt na displeji je také reprezentován „faceplatem“. Otevřeme ho poklikem myši na daném objektu.

TAG	AUTO	PCC	EQUIPMENT
XV2240	<input type="checkbox"/>	<input type="checkbox"/>	
PM2205	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
H2210	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
LV2201	<input type="checkbox"/>	<input type="checkbox"/>	
LIC2201	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
XV2250	<input type="checkbox"/>	<input type="checkbox"/>	

Obr. 38 Displej zobrazující stavy jednotlivých prvků

Spuštění sekvence

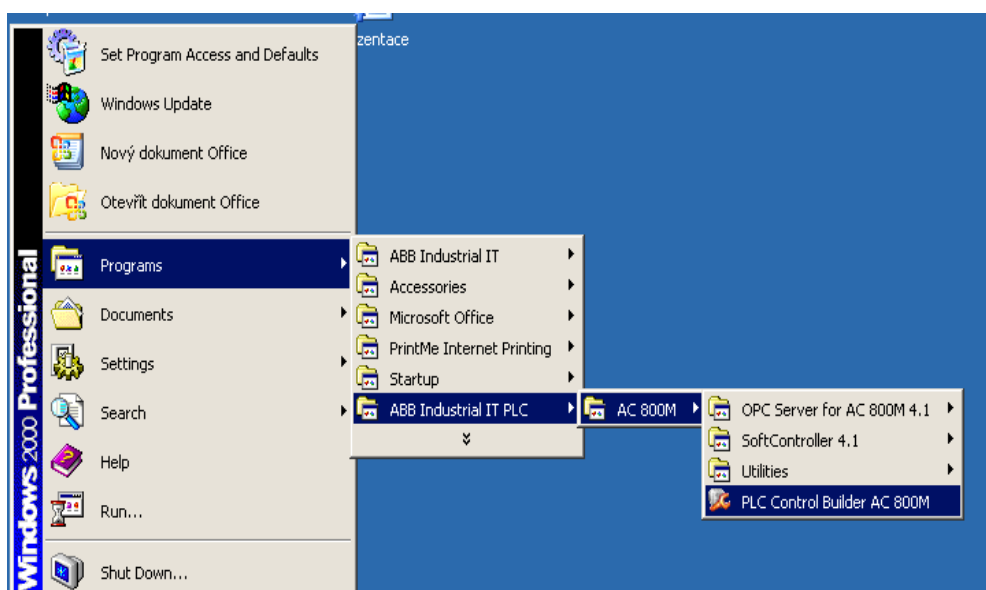
Před spuštěním procesu je nutno potvrdit všechny alarmy zobrazené v „Alarm Listu“. Různé barvy alarmů a čísel závažnosti indikují důležitost alarmů.

Ke startu sekvence slouží tlačítko “I” na “SEQ Faceplate”. Krok 2 v sekvenci nastavuje všechny objekty do automatického módu. Další krok otevírá přítokový ventil XV2240. Krok 4 slouží ke startu pumpy. Pumpa se rozběhne a začne plnit nádobu. K simulaci manuálního vypouštění slouží ventil HV2260 a k jeho otevření slouží položka „checkbox Field Feedback“. Sekvence může být kdykoli pozastavena stisknutím “Stop” tlačítka.

6 Návod pro práci s modelem ABB 800xA

Přihlášení do systému

Po naběhnutí operačního systému počítače je nutno chvíli počkat, až se zinicilizují všechny jeho součásti. Poté rozklikneme na ploše v dolním menu nabídku start a v programech vybereme odkaz „ABB Industrial IT PLC“ a spustíme v něm softwarové rozhraní „PLC Control Builder AC 800M“ přesně tak, jak je možno vidět na obr. 39.



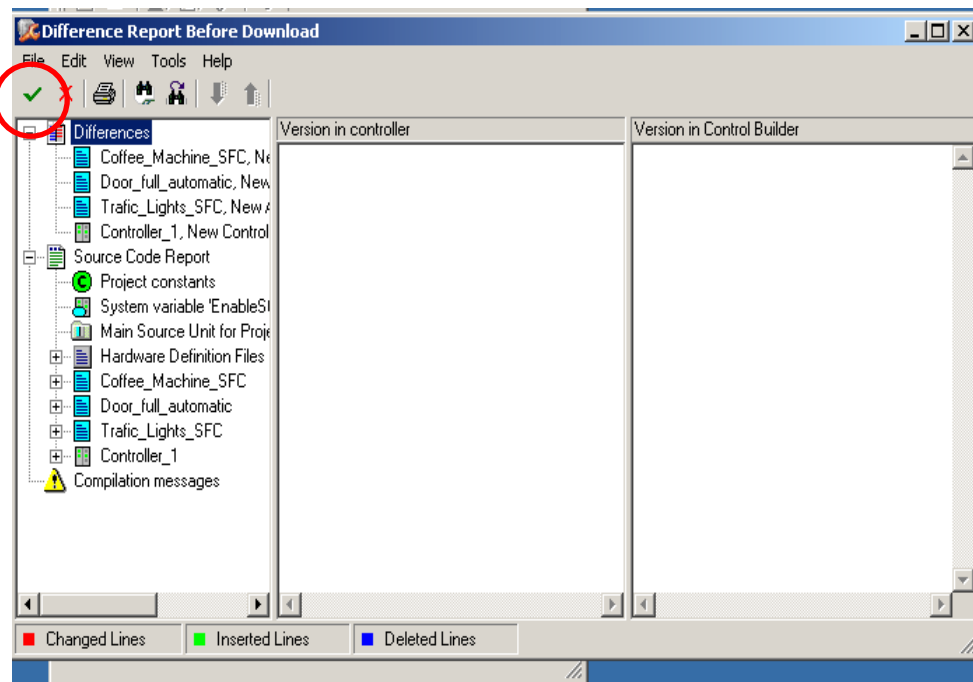
Obr. 39 Spuštění softwaru „PLC Control Builde AC 800M“

Spuštění a nahrání projektu do kontroleru

Pokud proběhlo vše z předcházejícího kroku v pořádku je nyní nutné nahrát náš projekt do kontroleru. V úvodní obrazovce „Control Builderu“ zvolíme v hlavním panelu pro otevření adresáře s naším projektem druhou ikonku zleva. Je také možno použít menu „File“ a tam otevřít projek pomocí položky „Open“ (C: / ABB Industrial IT Data / Engineer IT Data / PLC Control Builder AC800M 4.1 / Projects / Demo_VSB / Demo_VSB.prj).

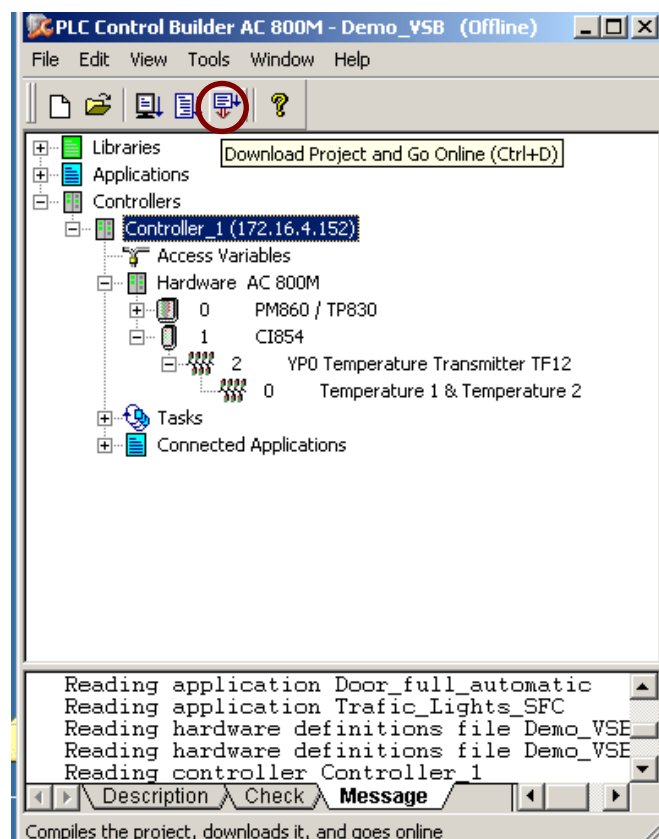
Pro nahrání dané aplikace do kontroleru je nutné zvolit pátou ikonku zleva v hlavním panelu (viz obr. 41).

Pokud je naše aplikace naprogramována pro ovládání na uživatelském panelu, tak se nám v průběhu jejího nahrávání do kontroleru objeví obrazovka „Diference Report Before Download“ (viz obr. 40). V její levé části máme zobrazeny jednotlivé naprogramované položky našeho projektu. Tyto je také nutno nahrát do ovládacího panelu. Slouží k tomu hned první ikonka zleva v hlavním panelu pod menu „File“, kterou je nutno potvrdit.



Obr. 40 Obrazovka pro nahrání aplikace do uživatelského panelu

Jestliže vše proběhlo úspěšně, měli bychom vidět v dolní části v okně zpráv kontroleru nápis „You are now online!“ (viz obr. 41).



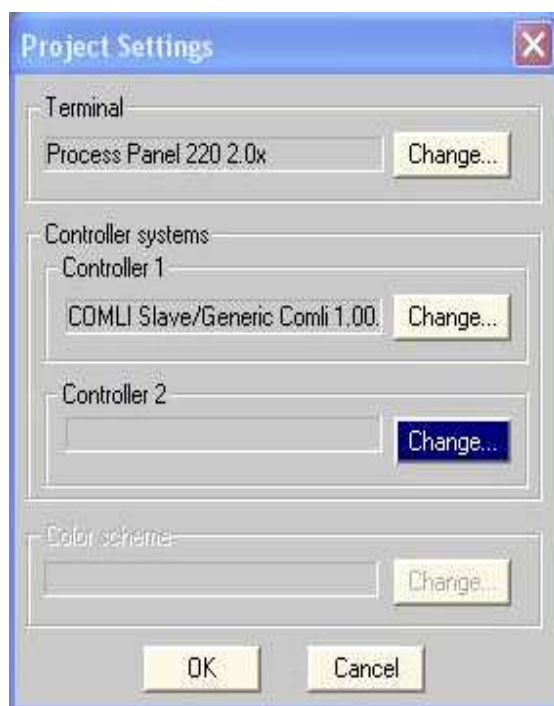
Obr. 41 Obrazovka projektu

Nahrání uživatelského rozhraní do ovládacího panelu

Pro ovládání dané aplikace přes uživatelský interface je nutno mít předem naprogramované zjednodušené vizualizační prostředí, ve kterém budeme poté pracovat a jehož prostřednictvím přes ovládací panel danou aplikaci ovládat. Programování této vizualizace se provádí v softwarovém rozhraní „Process Panel Builder“, kterému jsme se již věnovali v kapitole 2.1.8.

Kromě vytvoření dané vizualizace slouží software „Process Panel Builder“ také k jejímu nahrání do uživatelského panelu. Pro jeho spuštění poklepeme na ploše na ikonku s názvem „Process Panel Builder“. Daný software se nám následovně spustí a v menu „File“ otevřeme námi již vytvořenou aplikaci. Poté můžeme tuto již přímo nahrát do panelu a to tak, že v menu „Transfer“ vybereme hned první položku „Project“. Následně nám vyskočí obrazovka s názvem „Project transfer“ (viz obr. 45). Zde pro spuštění nahrávání aplikace klikneme na tlačítko „Send“.

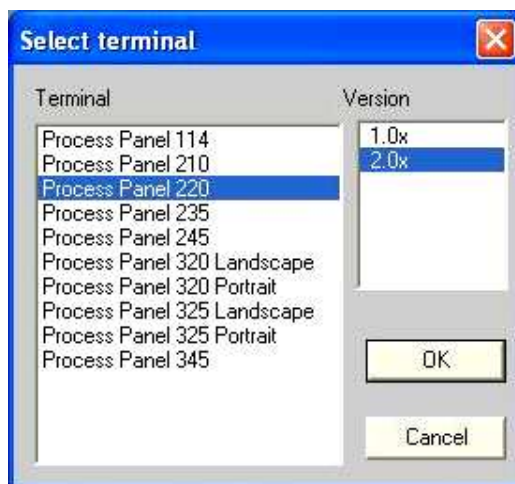
Pokud bychom chtěli vytvořit pro panel aplikaci úplně novou, tak zvolíme v menu „File“ položku „New“. Na úvod se nám objeví okno s názvem „Project Settings“ (viz obr. 42).



Obr. 42 Obrazovka nastavení „Project Settings“

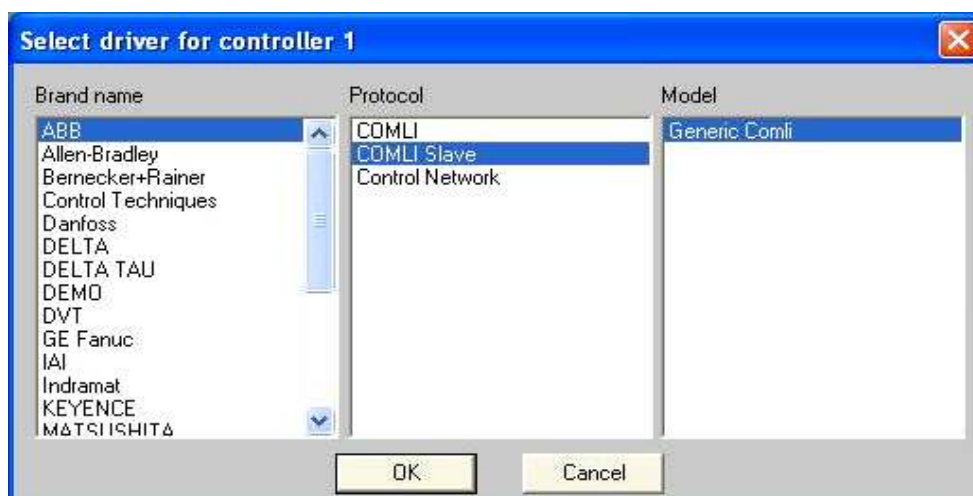
Toto okno obsahuje pod sebou tři textová pole. První z nich se nazývá „Terminal“, u kterého nám po kliknutí na tlačítko „Change“ vyskočí okno „Select terminal“, kde můžeme

nastavit jaký typ uživatelského panelu budeme využívat (viz obr. 43). V našem případě se jedná o „Process Panel 220“ verze „2.0x“. Následně potvrdíme tlačítkem „ok“.



Obr. 43 Nastavení typu uživatelského panelu

V dalších dvou textových polích se nastavuje druh komunikace mezi PC a ovládacím panelem. Pro tuto komunikaci je použit protokol „COMLI Slave“. Nastavíme jej v okně „Select driver for controller 1“, které se nám zobrazí opět po kliknutí na tlačítko „Change“ (viz obr. 44). Nastavení potvrdíme poté tl. „ok“. Jelikož má uživatelský interface pouze jeden výstup pro tuto komunikaci, nastavuje se příslušný protokol pouze v prvním textovém poli (viz obr. 42). Před nahráním dané vizualizace je nutno zkontrolovat, zda náhodou není panel připojen k PLC, které také využívá stejný komunikační protokol. Pokud ano, je zapotřebí od něj panel odpojit a připojit jej k PC přes sériovou linku.

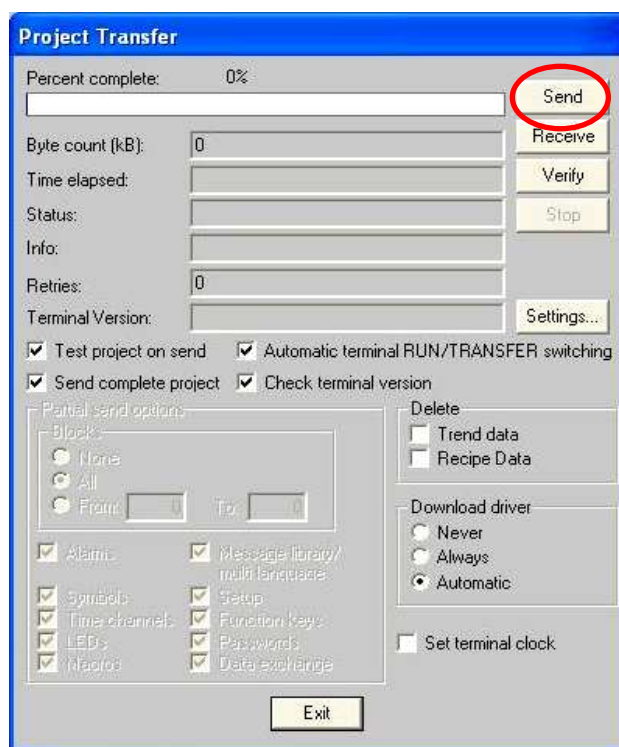


Obr. 44 Okno pro nastavení komunikačního protokolu

Pokud máme vše nastaveno, tak jak potřebujeme, potvrdíme toto nastavení v okně „Project Settings“ a následně se nám zobrazí základní bloková struktura námi již naprogramované vizualizace.

Tuto nahrajeme do uživatelského interface úplně stejným způsobem, jak už bylo popsáno na začátku u již vytvořené aplikace.

Daná úloha se nám poté nahraje do panelu a po jeho opětovném připojení k PLC můžeme příslušnou aplikaci bez problémů ovládat.



Obr. 45 Obrazovka pro nahrání aplikace do panelu

7 Závěr

V první části mé práce jsem se patřičně seznámil s oběma demy od firmy ABB a to jak s demem 800xA, tak i s demem dávkovacího systému. Podrobně jsem poté popsal jejich hardwarové prvky a softwarové vybavení. Co se týká právě programového vybavení, tak obě dema využívají technologii softwaru „Control Builder M“, kterou jsem popsal v kapitole 2.3.

V části druhé jsem se věnoval nejdůležitějšímu prvku tzv. „srdci“ samotných dem a to programovatelnému logickému automatu AC 800M. Zabýval jsem se konkrétně jeho hardwarovou strukturou a způsobem jeho programování. Zjistil jsem, že lze dané PLC programovat několika způsoby. Například pomocí blokových či liniových schémat nebo přímo programovacími algoritmy. Tyto tři způsoby se dají různě kombinovat, což považuji za velkou výhodu. Zajímavá pro mě velmi byla i komunikační struktura PLC a to především možnosti jejího rozšíření o přidání různých zásuvných modulů.

V dalším průběhu jsem se již věnoval zpracování praktické části mé práce. Bylo nutno odstranit komunikační závadu mezi ovládacím panelem a PLC u dema 800xA, způsobenou chybně nahanou aplikací v PLC.

K vyřešení tohoto problému a k následnému ověření funkčnosti komunikace jsem měl k dispozici již vytvořený projekt „Demo VŠB“, který jsem nahrál do PLC prostřednictvím již známého programového rozhraní „Control Builder“. K tomuto projektu byla vytvořena i jednoduchá vizualizace pro jeho ovládání přes uživatelský panel. Tuto jsem do něj nahrál prostřednictvím softwarového rozhraní „Process Panel Builder“. Když byla daná vizualizace již v panelu nahrána, tak se po chvíli tento s PLC zinicizoval a komunikace poté fungovala bez jakýchkoli problémů.

Funkčnost celého dema 800xA jsem poté ověřil na aplikaci řízení kolejové dopravy, kterou jsem přidal k již vytvořenému projektu „Demo VŠB“. Přidání pouhé aplikace proběhlo bez větších obtíží, ale co se týká jejího komunikačního rozhraní zde vznikl problém. Toto už nebylo možno pouze a jednoduše přidat, ale bylo ho nutno celé nastavit do komunikačního rozhraní projektu „Dema VŠB“ znovu od začátku. A to z toho důvodu, že PLC dema 800xA neumí pracovat se dvěma komunikačními rozhraními najednou. Po celkovém a zdlouhavém nastavování této aplikace do projektu vše nakonec dopadlo úspěšně. Celý projekt i s nově přidanou aplikací na řízení kolejové dopravy jsem poté nahrál do PLC.

Jeho funkčnost (dema) jsem již poté odzkoušel prostřednictvím nově přidané aplikace na laboratorním modelu kolejové trati.

V poslední části práce jsem vytvořil podrobné a jednoduché návody pro práci s oběma laboratorními modely dem a jejich uvedení do bezchybného provozu. Návody jsem doplnil o kopie obrazovek jednotlivých programů a jejich složek pro názornou ilustraci.

Co se týká směru práce do budoucna v dalším studiu, tak bych se chtěl věnovat experimentálním pokusům s elektromotorem pomocí frekvenčního měniče u dema 800xA. Dále vytvořit vizualizaci aplikace na řízení kolejové dopravy pro ovládací panel takéž u tohoto laboratorního modelu.

Použitá literatura

ABB, 2005A. *AC 800M Controller Hardware*. 308 stran

ABB, 2005B. *Modules and Termination Units*. 578 stran

ČSN ISO 690. *Bibliografické citace. Obsah, forma a struktura*. Praha: Český normalizační institut, 1996. 32 s.

FARANA, R. AJ. *Zpracování závěrečných textů z oblasti automatizace a informatiky*. Včetně anglicko-českého slovníku automatizační techniky a řízení. 1. vyd. Ostrava: VŠB-TU Ostrava, 2004. 116 s. ISBN 80-248-0557-X.

MARTINÁSKOVÁ, M. & ŠMEJKAL, L. *PLC a automatizace 1, základní pojmy, úvod do programování*. Praha: BEN, 2002. 224 s. ISBN 80-86056-58-9.

ŠMEJKAL, L. *PLC a automatizace 2, sekvenční logické systémy a základy fuzzy logiky*. Praha: BEN, 2005. 208 s. ISBN 80-73000-87-3.

MAREK, L. & HABERNAL, T. *Programovatelný automat ABB*, Ostrava: katedra automatizační techniky a řízení, VŠB-TU Ostrava, 2006. 16 stran.

UNGER, O. *Laboratorní úlohy řízení provozu na modelovém kolejišti*. Ostrava: katedra automatizační techniky a řízení, VŠB-TU Ostrava, 2008. 40 stran. Bakalářská práce, vedoucí: Pavlas, R.

BUBLÍK, D. *Elektronická podpora výuky předmětu Automatické řízení I*. Ostrava: katedra automatizační techniky a řízení, VŠB-TU Ostrava, 2008. 49 stran. Bakalářská práce, vedoucí: Wagnerová, R.

KÝVALA, H. *Řízení provozu na modelovém kolejišti*. Ostrava: katedra automatizační techniky a řízení, VŠB-TU Ostrava, 2008. 79 stran. Diplomová práce, vedoucí: Pavlas, R.

HABERNAL, T. *Laboratorní úlohy řízení provozu na modelovém kolejišti*. Ostrava: katedra automatizační techniky a řízení, VŠB-TU Ostrava, 2006. 56 stran. Bakalářská práce, vedoucí: Pavlas, R.

JEŽEK F. *Vytvoření distribuované řídicí aplikace pomocí automatizačních systémů ABB*, Ostrava: katedra měřicí a řídicí techniky, VŠB-TU Ostrava, 2007. 49 stran. Diplomová práce, vedoucí: Koziolek, J.

ŠOFER, P. *Využití SCADA/MMI systému pro podporu laboratorních měření*. Ostrava: katedra automatizační techniky a řízení, VŠB-TU Ostrava, 2008. 58 stran. Bakalářská práce, vedoucí: Škuta, J.

ABB *Firmware manual ACS 800*, 250 stran [PDF]

ABB *Hardware manual ACS 800*, 104 stran [PDF]

ABB 2005E. *AC 800M Controller Hardware - Hardware and Operation*. [online]. Västerås (Sweden): Automation Technology Products, Květen 2005 [cit. 2007-08-03]. číslo dokumentu: 3BSE036351R4101. Dostupné z www: <3BSE036351R4101_AC800M_HWOp.pdf > str. 25-34, 215-220

CENDELÍN, J. 2003. *Historie programovatelných automatů a jejich současné efektivní použití*. [online]. Plzeň: Západočeská universita. [cit. 2007-07-24]. Dostupné z www: <<http://www.odbornecasopisy.cz/automa/2003/au060306.htm>>

ABB 2005B. *OPC Server for AC 800M – Installation and Configuration*. [online]. Västerås (Sweden): Automation Technology Products, Květen 2005 [cit. 2007-08-04]. číslo dokumentu: 3BSE035983R4101. < URL: 3BSE035983R4101_CIO_OPCCServ.pdf >

ABB 2002B. *AC 800M Controller – The modular and scalable process controller*. [online]. Västerås (Sweden): Automation Technology Products, 2002 [cit. 2006-04-03]. číslo dokumentu: 3BSE 023 957 R0201. Dostupné z www: < URL: AC+800M.pdf >